Journal of Machine Learning Research vol (2012) 1–27

# Nonparametric Bayesian Mixed-effect Model: a Sparse Gaussian Process Approach

**Yuyang Wang** YWANG02@CS.TUFTS.EDU
**Roni Khardon** RONI@CS.TUFTS.EDU
*Department of Computer Science, Tufts University, Medford, MA 02155, USA*



## Abstract

Multi-task learning models using Gaussian processes (GP) have been developed and successfully applied in various applications. The main difficulty with this approach is the computational cost of inference using the union of examples from all tasks. Therefore sparse solutions, that avoid using the entire data directly and instead use a set of informative "representatives" are desirable. The paper investigates this problem for the grouped mixed-effect GP model where each individual response is given by a fixed-effect, taken from one of a set of unknown groups, plus a random individual effect function that captures variations among individuals. Such models have been widely used in previous work but no sparse solutions have been developed. The paper presents the first sparse solution for such problems, showing how the sparse approximation can be obtained by maximizing a variational lower bound on the marginal likelihood, generalizing ideas from single-task Gaussian processes to handle the mixed-effect model as well as grouping. Experiments using artificial and real data validate the approach showing that it can recover the performance of inference with the full sample, that it outperforms baseline methods, and that it outperforms state of the art sparse solutions for other multi-task GP formulations.

**Keywords:** Multi-task Learning, Gaussian Processes, Sparse Model, Mixed-effect Model

## 1. Introduction

In many real world problems we are interested in learning multiple tasks while the training set for each task is quite small. When the different tasks are related one can learn all tasks simultaneously and aim to get improved predictive performance by taking advantage of the common aspects of all tasks. This general idea is known as multi-task learning and it has been successfully investigated in several technical settings, with applications in many areas including medical diagnosis (Bi et al., 2008), recommendation systems (Dinuzzo et al., 2008) and HIV Therapy Screening (Bickel et al., 2008).

In this paper we explore Bayesian models especially using Gaussian Processes (GP) where sharing the prior and its parameters among the tasks can be seen to implement multi-task learning (Álvarez et al., 2011; Bonilla et al., 2008; Xue et al., 2007; Gelman, 2004; Yu et al., 2005; Schwaighofer et al., 2005; Pillonetto et al., 2010). Our focus is on the *functional mixed-effect model* (Lu et al., 2008; Pillonetto et al., 2010) where each task is modeled as a sum of a fixed effect shared by all the tasks and a random effect that can be interpreted as representing task specific deviations. In particular, both effects are realizations of zero-mean Gaussian processes. Thus, in this model, tasks share structure through hyperparameters of the prior and through the fixed effect portion. This model has shown success

in several applications, including geophysics (Lu et al., 2008), medicine (Pillonetto et al., 2010) and astrophysics (Wang et al., 2010). One of the main difficulties with this model, however, is computational cost, because while the number of samples per task $N_j$ is small, the total sample size $\sum_j N_j$ can be large, and the typical cubic complexity of GP inference can be prohibitively large (Yu et al., 2005). Some improvement can be obtained when all the input tasks share the same sampling points, or when different tasks share many of the input points (Pillonetto et al., 2009, 2010). However, if the number of distinct sampling points is large the complexity remains high. For example, this is the case in (Wang et al., 2010) where sample points are clipped to a fine grid to avoid the high cardinality of the example set.

The same problem, handling large samples, has been addressed in single task formalizations of GP, where several approaches for so-called sparse solutions have been developed (Rasmussen and Williams, 2005; Seeger and Lawrence, 2003; Snelson, 2006; Titsias, 2009). These methods approximate the GP with $m \ll N$ support variables (or inducing variables, pseudo inputs) $\mathcal{X}_m$ and their corresponding function values $\mathbf{f}_m$ and perform inference using this set.

In this paper, we develop a sparse solution for multi-task learning with GP in the context of the functional mixed effect model. Specifically, we extend the approach of Titsias (2009) and develop a variational approximation that allows us to efficiently learn the shared hyper-parameters and choose the sparse pseudo samples. In addition, we show how the variational approximation can be used to perform prediction efficiently once learning has been performed. Our approach is particularly useful when individual tasks have a small number of samples, different tasks do not share sampling points, and there is a large number of tasks. Our experiments, using artificial and real data, validate the approach showing that it can recover the performance of inference with the full sample, that it performs better than simple sparse approaches for multi-task GP, and that for some applications it significantly outperforms alternative sparse multi-task GP formulation (Álvarez and Lawrence, 2011).

To summarize, our contribution is threefold. First we introduce the first sparse solution for the multi-task GP in mixed-effect model. Second, we develop a variational model-selection approach for the proposed sparse model. Finally we evaluate the algorithm and several baseline approaches for multi-task GP, showing that the proposed method performs well.

This paper is organized as follows. Section 2 reviews the mixed-effect GP model and its direct inference. Section 3 develops the variational inference and model selection for the sparse mixed-effect GP model. Section 4 shows how to extend the sparse solution to the grouped mixed-effect GP model. We discuss related work in Section 5 and demonstrate the performance of the proposed approach using three datasets in Section 6. Section 7 concludes with a summary and directions for future work.

## 2. Mixed-effect GP for Multi-task Learning

In this section and the next one, we develop the mixed-effect model and its sparse solution without considering grouping. The model and results are extended to include grouping in Section 4. Consider a set of $M$ tasks where the data for the $j$th task is given by $\mathcal{D}^j = \{(\boldsymbol{x}_i^j, y_i^j)\}, i = 1, 2, \cdots, N_j$. Multi-task learning aims to learn all tasks simultane-

ously, taking the advantage of common aspects of different tasks. In this paper, given data $\breve{\boldsymbol{y}} = \{\mathcal{D}^j\}$, we are interested in learning the nonparametric Bayesian mixed-effect model and using the model to perform inference. The model captures each task $f^j$ as a sum of an average effect function and an individual variation specific to the $j$th task. More precisely (Pillonetto et al., 2010):

**Assumption 1** *For each $j$ and $\boldsymbol{x} \in \mathcal{X} \subset \mathbb{R}^d$,*

$$f^j(\boldsymbol{x}) = \bar{f}(\boldsymbol{x}) + \tilde{f}^j(\boldsymbol{x}), \quad j = 1, \cdots, M \tag{1}$$

*where $\bar{f}$ and $\{\tilde{f}^j\}$ are zero-mean Gaussian processes. In addition, $\bar{f}$ and the set of $\{\tilde{f}^j\}$ are assumed to be mutually independent with covariance functions $K(\cdot,\cdot)$ and $\widetilde{K}(\cdot,\cdot)$ respectively.*

Assumption 1 implies that for $j, l \in \{1, \cdots, M\}$, the following holds:

$$\mathbf{Cov}[f^j(\boldsymbol{s}), f^l(\boldsymbol{t})] = \mathbf{Cov}[\bar{f}(\boldsymbol{s}), \bar{f}(\boldsymbol{t})] + \delta_{jl} \cdot \mathbf{Cov}[\tilde{f}(\boldsymbol{s}), \tilde{f}(\boldsymbol{t})] \tag{2}$$

where $\delta_{jl}$ is the Kronecker delta function. Let $\breve{\boldsymbol{x}}$ be the concatenation of the examples from all tasks $\breve{\boldsymbol{x}} = (\boldsymbol{x}_i^j)$, and similarly let $\breve{\boldsymbol{y}} = (\boldsymbol{y}_i^j)$, where $i = 1, 2, \cdots, N_j, j = 1, 2, \cdots, M$ and $N = \sum_j N_j$. It can easily been seen that, for any $j \in \{1, \cdots, M\}$ and new input $\boldsymbol{x}^*$ for task $j$, we have

$$\begin{bmatrix} f^j(\boldsymbol{x}^*) \\ \breve{\boldsymbol{y}} \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} \mathbf{C}^\dagger(\boldsymbol{x}^*, \boldsymbol{x}^*) & \mathbf{C}^\dagger(\boldsymbol{x}^*, \breve{\boldsymbol{x}}) \\ \mathbf{C}^\dagger(\breve{\boldsymbol{x}}, \boldsymbol{x}^*) & \mathbf{C}^\dagger(\breve{\boldsymbol{x}}, \breve{\boldsymbol{x}}) + \sigma^2\mathbb{I} \end{bmatrix}\right) \tag{3}$$

where the covariance matrix $\mathbf{C}^\dagger$ is given by

$$\mathbf{C}^\dagger((\boldsymbol{x}_i^j), (\boldsymbol{x}_k^l)) = K(\boldsymbol{x}_i^j, \boldsymbol{x}_k^l) + \delta_{jl} \cdot \widetilde{K}((\boldsymbol{x}_i^j, \boldsymbol{x}_k^l).$$

From (3) we can extract the marginal distribution $\Pr(\breve{\boldsymbol{y}})$ where

$$\breve{\boldsymbol{y}} | \breve{\boldsymbol{x}} \sim \mathcal{N}(\mathbf{0}, \mathbf{C}^\dagger(\breve{\boldsymbol{x}}, \breve{\boldsymbol{x}}) + \sigma^2\mathbb{I}), \tag{4}$$

which can be used for model selection, that is, learning the hyper-parameters of the GP. (3) also provides the predictive distribution where

$$\begin{aligned} \mathbb{E}(f^j(\boldsymbol{x}^*)) &= \mathbf{C}^\dagger(\boldsymbol{x}^*, \breve{\boldsymbol{x}})(\mathbf{C}^\dagger(\breve{\boldsymbol{x}}, \breve{\boldsymbol{x}}) + \sigma^2\mathbb{I})^{-1}\breve{\boldsymbol{y}} \\ \mathbf{Cov}(f^j(\boldsymbol{x}^*)) &= \mathbf{C}^\dagger(\boldsymbol{x}^*, \boldsymbol{x}^*) - \mathbf{C}^\dagger(\boldsymbol{x}^*, \breve{\boldsymbol{x}})(\mathbf{C}^\dagger(\breve{\boldsymbol{x}}, \breve{\boldsymbol{x}}) + \sigma^2\mathbb{I})^{-1}\mathbf{C}^\dagger(\breve{\boldsymbol{x}}, \boldsymbol{x}^*). \end{aligned} \tag{5}$$

This works well in that sharing the information improves predictive performance but, as the number of tasks grows, the dimension $N$ increases leading to slow inference scaling as $\mathcal{O}(N^3)$. In other words, even though each task may have a very small sample, the multi-task inference problem becomes infeasible when the number of tasks is large.

In single task GP regression, to reduce the computational cost, several sparse GP approaches have been proposed (Rasmussen and Williams, 2005; Seeger and Lawrence, 2003; Snelson, 2006; Titsias, 2009). In general, these methods approximate the GP with a small number $m \ll N$ of support variables and perform inference using this subset and the corresponding function values $\mathbf{f}_m$. Different approaches differ in how they choose the support

variables and the simplest approach is to choose a random subset of the given data points. Recently, Titsias (2009) introduced a sparse method based on variational inference using a set $\mathcal{X}_m$ of inducing samples, which are different from the training points. In this approach, the sample points $\mathcal{X}_m$ are chosen to maximize a variational lower bound on the marginal likelihood, therefore providing a clear methodology for the choice of the support set. Following their idea, Álvarez et al. (2010) proposed the variational inference for sparse convolved multiple output GPs.

In this paper we extend this approach to provide a sparse solution for the aforementioned model as well as generalizing it to the Grouped mixed-effect GP model (Wang et al., 2010). As in the case of sparse methods for single task GP, the key idea is to introduce a small set of $m$ auxiliary inducing sample points $\mathcal{X}_m$ and base the learning and inference on these points. For the multi-task case, each $\tilde{f}^j(\cdot)$ is specific to the $j$th task. Therefore, it makes sense to induce values only for the fixed-effect portion $\mathbf{f}_m = \bar{f}(\mathcal{X}_m)$. The details of this construction are developed in the following sections.

## 3. Sparse mixed-effect GP Model

In this section, we develop a sparse solution for the mixed-effect model without group effect. The model is simpler to analyze and apply, and it thus provides a good introduction to the results developed in the next section for the grouped model.

### 3.1. Variational Model Selection

In this section we specify the sparse model, and show how we can learn the hyper-parameters and the inducing variables using the sparse model. As mentioned above, we introduce auxiliary inducing sample points $\mathcal{X}_m$ and hidden variables $\mathbf{f}_m = \bar{f}(\mathcal{X}_m)$. Let $\mathbf{f}^j = \bar{f}(\boldsymbol{x}^j) \in \mathbb{R}^{N_j}$ and $\tilde{\mathbf{f}}^j = \tilde{f}(\boldsymbol{x}^j) \in \mathbb{R}^{N_j}$ denote the values of the two functions at $\boldsymbol{x}^j$. In addition let $\mathbf{C}_{*j} = \mathbf{C}(\boldsymbol{x}^*, \boldsymbol{x}^j)$, $\mathbf{C}_{jj} = \mathbf{C}(\boldsymbol{x}^j, \boldsymbol{x}^j)$ and $\mathbf{C}_{mm} = \mathbf{C}(\mathcal{X}_m, \mathcal{X}_m)$, and similarly for $\widetilde{\mathbf{C}}_{*j}, \widetilde{\mathbf{C}}_{jj}, \widetilde{\mathbf{C}}_{mm}$.

To learn the hyper-parameters we wish to maximize the marginal likelihood $\Pr(\breve{\boldsymbol{y}})$ where $\breve{\boldsymbol{y}}$ is all the observations. In the following we develop a variational lower bound for this quantity. To this end, we need the complete data likelihood and the variational distribution.

- The complete data likelihood $\Pr(\{\boldsymbol{y}^j\}, \{\mathbf{f}^j, \tilde{\mathbf{f}}^j\}, \mathbf{f}_m)$ is given by:

$$\begin{aligned}&\Pr(\{\boldsymbol{y}^j\}|\{\mathbf{f}^j, \tilde{\mathbf{f}}^j\}) \Pr(\{\tilde{\mathbf{f}}^j\}) \Pr(\{\mathbf{f}^j\}|\mathbf{f}_m) \Pr(\mathbf{f}_m) \\ &= \left[\prod_{j=1}^{M} \Pr(\boldsymbol{y}^j|\mathbf{f}^j, \tilde{\mathbf{f}}^j) \Pr(\tilde{\mathbf{f}}^j)\right] \Pr(\{\mathbf{f}^j\}|\mathbf{f}_m) \Pr(\mathbf{f}_m).\end{aligned}$$

- We approximate the posterior $\Pr(\{\mathbf{f}^j, \tilde{\mathbf{f}}^j\}, \mathbf{f}_m|\{\boldsymbol{y}^j\})$ on the hidden variables by

$$q(\{\mathbf{f}^j, \tilde{\mathbf{f}}^j\}, \mathbf{f}_m) = \left[\prod_{j=1}^{M} \Pr(\tilde{\mathbf{f}}^j|\mathbf{f}^j, \boldsymbol{y}^j)\right] \Pr(\{\mathbf{f}^j\}|\mathbf{f}_m)\phi(\mathbf{f}_m) \tag{6}$$

  which extends the variational form used by Titsias (2009) to handle the individual variations as well as the multiple tasks. One can see that the variational distribution

is not completely in free form. Instead, $q(\cdot)$ preserves the exact form of $\Pr(\tilde{\mathbf{f}}^j|\mathbf{f}^j,\boldsymbol{y}^j)$ and in using $\Pr(\{\mathbf{f}^j\}|\mathbf{f}_m)$ it implicitly assumes that $\mathbf{f}_m$ is a sufficient statistic for $\{\mathbf{f}^j\}$. The free form $\phi(\mathbf{f}_m)$ corresponds to $\Pr(\mathbf{f}_m|\breve{\boldsymbol{y}})$ but allows it to diverge from this value to compensate for the assumption that $\mathbf{f}_m$ is sufficient. Notice that we are not making any assumption about the sufficiency of $\mathbf{f}_m$ in the generative model and the approximation is entirely due to the variational distribution. An additional assumption is added later to derive a simplified form of the predictive distribution.

With the two ingredients ready, the variational lower bound (Jordan et al., 1999; Bishop, 2006), denoted as $F_V(\mathcal{X}_m,\phi)$, is given by:

$$\begin{aligned}
\Pr(\breve{\boldsymbol{y}}) &\geqslant F_V(\mathcal{X}_m,\phi)\\
&= \int q(\{\mathbf{f}^j,\tilde{\mathbf{f}}^j\},\mathbf{f}_m)\times\log\left[\frac{\Pr(\{\boldsymbol{y}^j\},\{\mathbf{f}^j,\tilde{\mathbf{f}}^j\},\mathbf{f}_m)}{q(\{\mathbf{f}^j,\tilde{\mathbf{f}}^j\},\mathbf{f}_m)}\right]d\{\mathbf{f}^j\}d\{\tilde{\mathbf{f}}^j\}d\mathbf{f}_m\\
&= \int\left[\prod_{j=1}^{M}\Pr(\tilde{\mathbf{f}}^j|\mathbf{f}^j,\boldsymbol{y}^j)\right]\Pr(\{\mathbf{f}^j\}|\mathbf{f}_m)\phi(\mathbf{f}_m)\\
&\quad\times\log\left[\prod_{l=1}^{M}\frac{\Pr(\boldsymbol{y}^l|\mathbf{f}^l,\tilde{\mathbf{f}}^l)\Pr(\tilde{\mathbf{f}}^l)}{\Pr(\tilde{\mathbf{f}}^l|\mathbf{f}^l,\boldsymbol{y}^l)}\cdot\frac{\Pr(\mathbf{f}_m)}{\phi(\mathbf{f}_m)}\right]d\{\mathbf{f}^j\}d\{\tilde{\mathbf{f}}^j\}d\mathbf{f}_m\\
&= \int\phi(\mathbf{f}_m)\left\{\log G(\mathbf{f}_m,\mathcal{Y})+\log\left[\frac{\Pr(\mathbf{f}_m)}{\phi(\mathbf{f}_m)}\right]\right\}d\mathbf{f}_m.
\end{aligned}$$

The inner integral denoted as $\log G(\mathbf{f}_m,\mathcal{Y})$ is

$$\begin{aligned}
&\int\left[\prod_{j=1}^{M}\Pr(\tilde{\mathbf{f}}^j|\mathbf{f}^j,\boldsymbol{y}^j)\right]\Pr(\{\mathbf{f}^j\}|\mathbf{f}_m)\times\sum_{l=1}^{M}\log\left[\frac{\Pr(\boldsymbol{y}^l|\mathbf{f}^l,\tilde{\mathbf{f}}^l)\Pr(\tilde{\mathbf{f}}^l)}{\Pr(\tilde{\mathbf{f}}^l|\mathbf{f}^l,\boldsymbol{y}^l)}\right]d\{\mathbf{f}^j\}d\{\tilde{\mathbf{f}}^j\}\\
&=\sum_{j=1}^{M}\int\Pr(\tilde{\mathbf{f}}^j|\mathbf{f}^j,\boldsymbol{y}^j)\Pr(\mathbf{f}^j|\mathbf{f}_m)\times\log\left[\frac{\Pr(\boldsymbol{y}^j|\mathbf{f}^j,\tilde{\mathbf{f}}^j)\Pr(\tilde{\mathbf{f}}^j)}{\Pr(\tilde{\mathbf{f}}^j|\mathbf{f}^j,\boldsymbol{y}^j)}\right]d\mathbf{f}^j d\tilde{\mathbf{f}}^j
\end{aligned}\tag{7}$$

where the second line holds because in the sum indexed by $l$ all the product measures

$$\prod_{j=1,j\neq l}^{M}\Pr(\tilde{\mathbf{f}}^j|\mathbf{f}^j,\boldsymbol{y}^j)\Pr(\{\mathbf{f}^n\}_{n\neq l}|\mathbf{f}_m,\mathbf{f}_l)d\{\mathbf{f}^j\}d\{\tilde{\mathbf{f}}^j\},$$

are integrated to 1, leaving only the $j$-th integral. In subsection 3.1 we show that

$$\log G(\mathbf{f}_m,\mathcal{Y})=\sum_{j=1}^{m}\left[\log\left[\mathcal{N}(\boldsymbol{y}^j|\boldsymbol{\alpha}_j,\widehat{\mathbf{C}}_{jj})\right]-\frac{1}{2}\mathbf{Tr}\left[(\mathbf{C}_{jj}-\boldsymbol{Q}_{jj})[\widehat{\mathbf{C}}_{jj}]^{-1}\right]\right]\tag{8}$$

where $\boldsymbol{\alpha}_j = \mathbf{C}_{jm}\mathbf{C}_{mm}^{-1}\mathbf{f}_m$, $\widehat{\mathbf{C}}_{jj} = \sigma_j^2\mathbb{I} + \widetilde{\mathbf{C}}_{jj}$, and $\boldsymbol{Q}_{jj} = \mathbf{C}_{jm}\mathbf{C}_{mm}^{-1}\mathbf{C}_{mj}$. Thus we have

$$\begin{aligned} F_V(\mathcal{X}_m, \phi) &= \int \phi(\mathbf{f}_m) \left[\log G(\mathbf{f}_m, \mathcal{Y}) + \log\left[\frac{\Pr(\mathbf{f}_m)}{\phi(\mathbf{f}_m)}\right]\right] d\mathbf{f}_m \\ &= \int \phi(\mathbf{f}_m) \log\left[\frac{\prod_j \left[\mathcal{N}(\boldsymbol{y}^j|\boldsymbol{\alpha}_j, \widehat{\mathbf{C}}_{jj})\right] \Pr(\mathbf{f}_m)}{\phi(\mathbf{f}_m)}\right] d\mathbf{f}_m \\ &\quad - \frac{1}{2}\sum_{j=1}^{M} \mathbf{Tr}\left[(\mathbf{C}_{jj} - \boldsymbol{Q}_{jj})[\widehat{\mathbf{C}}_{jj}]^{-1}\right]. \end{aligned} \tag{9}$$

Let $\boldsymbol{v}$ be a random variable and $g$ any function, then by Jensen's inequality $\mathbb{E}[\log g(\boldsymbol{v})] \leq \log \mathbb{E}[g(\boldsymbol{v})]$. Therefore, the best lower bound we can derive from (9), if it is achievable, is the case where equality holds in Jensen's inequality. In subsection 3.2 we show that $\phi(\mathbf{f}_m)$ can be chosen to obtain equality, and therefore, the variational lower bound is

$$F_V(\mathcal{X}_m, \phi) = \log \int \prod_j \left[\mathcal{N}(\boldsymbol{y}^j|\boldsymbol{\alpha}_j, \widehat{\mathbf{C}}_{jj})\right] \Pr(\mathbf{f}_m) d\mathbf{f}_m - \frac{1}{2}\sum_{j=1}^{M} \mathrm{Tr}\left[(\mathbf{C}_{jj} - Q_{jj})[\widehat{\mathbf{C}}_{jj}]^{-1}\right].$$

Evaluating the integral by marginalizing out $\mathbf{f}_m$ and recalling that $\breve{\boldsymbol{y}}$ is the concatenation of the $\boldsymbol{y}^j$, we get

$$F_V(\mathcal{X}_m, \boldsymbol{\theta}, \tilde{\boldsymbol{\theta}}) = \log\left[\mathcal{N}(\breve{\boldsymbol{y}}|\mathbf{0}, \boldsymbol{\Lambda}_m\mathbf{C}_{mm}^{-1}\boldsymbol{\Lambda}_m^T + \widehat{\mathbf{C}}^m)\right] - \sum_{j=1}^{m}\left[\frac{1}{2}\mathbf{Tr}\left[(\mathbf{C}_{jj} - \boldsymbol{Q}_{jj})[\widehat{\mathbf{C}}_{jj}]^{-1}\right]\right] \tag{10}$$

where

$$\boldsymbol{\Lambda}_m = \begin{pmatrix} \mathbf{C}_{1m} \\ \mathbf{C}_{2m} \\ \vdots \\ \mathbf{C}_{Mm} \end{pmatrix} \quad \text{and} \quad \widehat{\mathbf{C}}^m = \bigoplus_{j=1}^{M} \widehat{\mathbf{C}}_{jj} = \begin{pmatrix} \widehat{\mathbf{C}}_{11} & & & \\ & \widehat{\mathbf{C}}_{22} & & \\ & & \ddots & \\ & & & \widehat{\mathbf{C}}_{MM} \end{pmatrix}.$$

Thus, we have explicitly written the parameters that can be chosen to further optimize the lower bound, namely the support inputs $\mathcal{X}_m$, and the hyper-parameters $\boldsymbol{\theta}$ and $\tilde{\boldsymbol{\theta}}$ in $K$ and $\widetilde{K}$ respectively. By calculating derivatives of (10) we can optimize the lower bound using a gradient based method. In the experiments in this paper, we use stochastic gradient descent (SGD), which works better than the conjugate gradient (CG) in this scenario where the number of tasks is large. Titsias (2009) outlines methods that can be used when gradients are not useful.

#### 3.1.1. Evaluating $\log G(\mathbf{f}_m, \mathcal{Y})$

Consider the $j$-th element in the sum of (7):

$$\begin{aligned}
\widehat{G}_j(\mathbf{f}^j, \boldsymbol{y}^j) &= \int \Pr(\tilde{\mathbf{f}}^j|\mathbf{f}^j, \boldsymbol{y}^j) \Pr(\mathbf{f}^j|\mathbf{f}_m) \log \left[ \frac{\Pr(\boldsymbol{y}^j|\mathbf{f}^j, \tilde{\mathbf{f}}^j) \Pr(\tilde{\mathbf{f}}^j)}{\Pr(\tilde{\mathbf{f}}^j|\mathbf{f}^j, \boldsymbol{y}^j)} \right] d\mathbf{f}^j d\tilde{\mathbf{f}}^j \\
&= \int \Pr(\tilde{\mathbf{f}}^j|\mathbf{f}^j, \boldsymbol{y}^j) \Pr(\mathbf{f}^j|\mathbf{f}_m) \\
&\quad \times \log \left[ \frac{\Pr(\tilde{\mathbf{f}}^j|\boldsymbol{y}^j, \mathbf{f}^j) \Pr(\boldsymbol{y}^j|\mathbf{f}^j)}{\Pr(\tilde{\mathbf{f}}^j|\mathbf{f}^j)} \cdot \frac{\Pr(\tilde{\mathbf{f}}^j)}{\Pr(\tilde{\mathbf{f}}^j|\mathbf{f}^j, \boldsymbol{y}^j)} \right] d\mathbf{f}^j d\tilde{\mathbf{f}}^j \\
&= \int \Pr(\mathbf{f}^j|\mathbf{f}_m) \log \left[ \Pr(\boldsymbol{y}^j|\mathbf{f}^j) \right] \left( \int \Pr(\tilde{\mathbf{f}}^j|\mathbf{f}^j, \boldsymbol{y}^j) d\tilde{\mathbf{f}}^j \right) d\mathbf{f}^j \\
&= \int \Pr(\mathbf{f}^j|\mathbf{f}_m) \log \left[ \Pr(\boldsymbol{y}^j|\mathbf{f}^j) \right] d\mathbf{f}^j = \mathbb{E}_{[\mathbf{f}^j|\mathbf{f}_m]} \log \left[ \Pr(\boldsymbol{y}^j|\mathbf{f}^j) \right]
\end{aligned}$$

where the third line holds because of the independence between $\tilde{\mathbf{f}}^j$ and $\mathbf{f}^j$. We next show how this expectation can be evaluated. This is more complex than the single-task case because of the coupling of the fixed-effect and the random effect.

Recall that

$$\Pr(\mathbf{f}^j|\mathbf{f}_m) = \mathcal{N}(\mathbf{f}^j|\mathbf{C}_{jm}\mathbf{C}_{mm}^{-1}\mathbf{f}_m, \mathbf{C}_{jj} - \mathbf{C}_{jm}\mathbf{C}_{mm}^{-1}\mathbf{C}_{mj})$$

and

$$\boldsymbol{y}^j|\mathbf{f}^j \sim \mathcal{N}(\mathbf{f}^j, \widehat{\mathbf{C}}_{jj})$$

where $\widehat{\mathbf{C}}_{jj} = \sigma^2\mathbb{I} + \widetilde{\mathbf{C}}_{jj}$. Denote $\widehat{\mathbf{C}}_{jj}^{-1} = \boldsymbol{L}^{\mathrm{T}}\boldsymbol{L}$ where $\boldsymbol{L}$ can be chosen as its Cholesky decomposition, we have

$$\begin{aligned}
\log \left[ \Pr(\boldsymbol{y}^j|\mathbf{f}^j) \right] &= -\frac{1}{2}(\boldsymbol{y}^j - \mathbf{f}^j)^T \widehat{\mathbf{C}}_{jj}^{-1}(\boldsymbol{y}^j - \mathbf{f}^j) + \log \left[ (2\pi)^{-\frac{N_j}{2}} \right] + \log \left[ |\widehat{\mathbf{C}}_{jj}|^{-\frac{1}{2}} \right] \\
&= -\frac{1}{2}(\boldsymbol{L}\boldsymbol{y}^j - \boldsymbol{L}\mathbf{f}^j)^T(\boldsymbol{L}\boldsymbol{y}^j - \boldsymbol{L}\mathbf{f}^j) + \log \left[ (2\pi)^{-\frac{N_j}{2}} \right] + \log \left[ |\widehat{\mathbf{C}}_{jj}|^{-\frac{1}{2}} \right].
\end{aligned}$$

Notice that

$$\Pr(\boldsymbol{L}\mathbf{f}^j|\mathbf{f}_m) = \mathcal{N}(\boldsymbol{L}\mathbf{C}_{jm}\mathbf{C}_{mm}^{-1}\mathbf{f}_m, \boldsymbol{L}(\mathbf{C}_{jj} - \boldsymbol{Q}_{jj})\boldsymbol{L}^{\mathrm{T}})$$

where $\boldsymbol{Q}_{jj} = \mathbf{C}_{jm}\mathbf{C}_{mm}^{-1}\mathbf{C}_{mj}$. Recall the fact that for $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ and a constant vector $\boldsymbol{a}$, we have $\mathbb{E}[\|\boldsymbol{a} - \boldsymbol{x}\|^2] = \|\boldsymbol{a} - \boldsymbol{\mu}\|^2 + \mathbf{Tr}(\boldsymbol{\Sigma})$. Thus,

$$\begin{aligned}
\mathbb{E}_{[\mathbf{f}^j|\mathbf{f}_m]} \log \left[ \Pr(\boldsymbol{y}^j|\mathbf{f}^j) \right] &= -\frac{1}{2}\|\boldsymbol{L}\boldsymbol{y}^j - \boldsymbol{L}\mathbf{C}_{jm}\mathbf{C}_{mm}^{-1}\mathbf{f}_m\|^2 \\
&\quad - \frac{1}{2}\mathbf{Tr}(\boldsymbol{L}(\mathbf{C}_{jj} - \boldsymbol{Q}_{jj})\boldsymbol{L}^T) + \log \left[ (2\pi)^{-\frac{N_j}{2}} \right] + \log \left[ |\widehat{\mathbf{C}}_{jj}|^{-\frac{1}{2}} \right] \\
&= \left\{ -\frac{1}{2} \left[ \boldsymbol{y} - \mathbf{C}_{jm}\mathbf{C}_{mm}^{-1}\mathbf{f}_m \right]^T \left( L^T L \right) \left[ \boldsymbol{y} - \mathbf{C}_{jm}\mathbf{C}_{mm}^{-1}\mathbf{f}_m \right] \right. \\
&\quad \left. + \log \left[ (2\pi)^{-\frac{N_j}{2}} \right] + \log \left[ |\mathbf{C}_{jj}|^{-\frac{1}{2}} \right] \right\} - \frac{1}{2}\mathrm{Tr} \left[ L(\mathbf{C}_{jj} - \boldsymbol{Q}_{jj})L^T \right] \\
&= \log \left[ \mathcal{N}(\boldsymbol{y}^j|\boldsymbol{\alpha}_j, \widehat{\mathbf{C}}_{jj}) \right] - \frac{1}{2}\mathbf{Tr} \left[ (\mathbf{C}_{jj} - \boldsymbol{Q}_{jj})\widehat{\mathbf{C}}_{jj}^{-1} \right]
\end{aligned}$$

where $\boldsymbol{\alpha}_j = \mathbf{C}_{jm}\mathbf{C}_{mm}^{-1}\mathbf{f}_m$. Finally, calculating $\sum_j \widehat{G}_j(\mathbf{f}^j, \boldsymbol{y}^j)$ we get (8).

#### 3.1.2. Variational distribution $\phi^*(\mathbf{f}_m)$

For equality to hold in Jensen's inequality, the function inside the log must be constant. In our case this is easily achieved because $\phi(\mathbf{f}_m)$ is a free parameter, and we can set

$$\left[\frac{\prod_j \left[\mathcal{N}(\boldsymbol{y}^j|\boldsymbol{\alpha}_j, \widehat{\mathbf{C}}_{jj})\right] \Pr(\mathbf{f}_m)}{\phi(\mathbf{f}_m)}\right] \equiv c,$$

yielding the bound given in (10). Setting $\phi(\mathbf{f}_m) \propto \prod_j \left[\mathcal{N}(\boldsymbol{y}^j|\boldsymbol{\alpha}_j, \widehat{\mathbf{C}}_{jj})\right] \Pr(\mathbf{f}_m)$ yields the form of the optimal variational distribution

$$\begin{aligned}\phi^*(\mathbf{f}_m) &\propto \prod_j \left[\mathcal{N}(\boldsymbol{y}^j|\boldsymbol{\alpha}_j, \widehat{\mathbf{C}}_{jj})\right] \Pr(\mathbf{f}_m) \\ &\propto \exp\left\{ -\frac{1}{2}\mathbf{f}_m^T \left[\mathbf{C}_{mm}^{-1}\boldsymbol{\Phi}\mathbf{C}_{mm}^{-1}\right]\mathbf{f}_m + \mathbf{f}_m^{\mathrm{T}}\left(\mathbf{C}_{mm}^{-1}\sum_j \mathbf{C}_{mj}\left[\widehat{\mathbf{C}}_{jj}\right]^{-1}\boldsymbol{y}^j\right)\right\},\end{aligned}$$

from which we observe that $\phi^*(\mathbf{f}_m)$ is

$$\mathcal{N}\left(\mathbf{f}_m \middle| \mathbf{C}_{mm}\boldsymbol{\Phi}^{-1}\sum_j \mathbf{C}_{mj}[\widehat{\mathbf{C}}_{jj}]^{-1}\boldsymbol{y}_j, \mathbf{C}_{mm}\boldsymbol{\Phi}^{-1}\mathbf{C}_{mm}\right) \tag{11}$$

where $\boldsymbol{\Phi} = \mathbf{C}_{mm} + \sum_j \mathbf{C}_{mj}[\widehat{\mathbf{C}}_{jj}]^{-1}\mathbf{C}_{jm}$. Notice that by choosing the number of tasks to be 1 and the random effect to be a noise process, i.e. $\widetilde{K}(s,t) = \sigma^2\delta(s,t)$, (10) and (26) are exactly the variational lower bound and the corresponding variational distribution in (Titsias, 2009).

### 3.2. Prediction using the Variational Solution

Given any task $j$, our goal is to calculate the predictive distribution of $f^j(\boldsymbol{x}^*) = \bar{f}(\boldsymbol{x}^*) + \tilde{f}^j(\boldsymbol{x}^*)$ at some new input point $\boldsymbol{x}^*$. As described before, the full inference is expensive and therefore we wish to use the variational approximation for the prediction as well. The key assumption is that $\mathbf{f}_m$ contains as much information as $\breve{\boldsymbol{y}}$ in terms of making prediction for $\bar{f}$. To start with, it is easy to see that the predictive distribution is Gaussian and that it satisfies

$$\begin{aligned}\mathbb{E}[f^j(\boldsymbol{x}^*)|\breve{\boldsymbol{y}}] &= \mathbb{E}[\bar{f}(\boldsymbol{x}^*)|\breve{\boldsymbol{y}}] + \mathbb{E}[\tilde{f}^j(\boldsymbol{x}^*)|\breve{\boldsymbol{y}}] \\ \mathbf{Var}[f^j(\boldsymbol{x}^*)|\breve{\boldsymbol{y}}] &= \mathbf{Var}[\bar{f}(\boldsymbol{x}^*)|\breve{\boldsymbol{y}}] + \mathbf{Var}[\tilde{f}^j(\boldsymbol{x}^*)|\breve{\boldsymbol{y}}] + 2\mathbf{Cov}[\bar{f}(\boldsymbol{x}^*)\tilde{f}^j(\boldsymbol{x}^*)|\breve{\boldsymbol{y}}].\end{aligned} \tag{12}$$

The above equation is more complex than the predictive distribution for single-task sparse GP because of the coupling induced by $\bar{f}(\boldsymbol{x}^*)\tilde{f}^j(\boldsymbol{x}^*)|\breve{\boldsymbol{y}}$. We next show how this can be calculated via conditioning.

To calculate the terms in (12), three parts are needed, i.e., $\Pr(\bar{f}(\boldsymbol{x}^*)|\breve{\boldsymbol{y}})$, $\Pr(\tilde{f}(\boldsymbol{x}^*)|\breve{\boldsymbol{y}})$ and $\mathbf{Cov}[\bar{f}(\boldsymbol{x}^*)\tilde{f}_j(\boldsymbol{x}^*)|\breve{\boldsymbol{y}}]$. Using the assumption of the variational form given in (6), we have the following facts,

1. $\mathbf{f}_m|\breve{\boldsymbol{y}} \sim \phi^*(\mathbf{f}_m) = \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{A})$ where $\boldsymbol{\mu}$ and $\boldsymbol{A}$ are given in (11).

2. $\mathbf{f}_m$ is sufficient for $\{\mathbf{f}^j\}$, i.e. $\Pr(\{\mathbf{f}^j\}|\mathbf{f}_m, \breve{\boldsymbol{y}}) = \Pr(\{\mathbf{f}^j\}|\mathbf{f}_m)$. Since we are interested in prediction for each task separately, by marginalizing out $\mathbf{f}^l, l \neq j$, we also have $\Pr(\mathbf{f}^j|\mathbf{f}_m, \breve{\boldsymbol{y}}) = \Pr(\mathbf{f}^j|\mathbf{f}_m)$ and

$$\mathbf{f}^j|\mathbf{f}_m, \breve{\boldsymbol{y}} \sim \mathcal{N}\left(\mathbf{C}_{jm}\mathbf{C}_{mm}^{-1}\mathbf{f}_m, \mathbf{C}_{jj} - \mathbf{C}_{jm}\mathbf{C}_{mm}^{-1}\mathbf{C}_{mj}\right). \tag{13}$$

3. For $\tilde{f}^j(\boldsymbol{x}^*)$ we can view $\boldsymbol{y}^j - \mathbf{f}^j$ as noisy realizations from the same GP as $\tilde{f}^j(\boldsymbol{x}^j)$ and therefore

$$\tilde{f}^j(\boldsymbol{x}^*)|\mathbf{f}^j, \breve{\boldsymbol{y}} \sim \mathcal{N}\left(\widetilde{\mathbf{C}}_{*j}\left[\widetilde{\mathbf{C}}_{jj} + \sigma_j^2\mathbb{I}_j\right]^{-1}\left[\boldsymbol{y}^j - \mathbf{f}^j\right], \widetilde{\mathbf{C}}_{**} - \widetilde{\mathbf{C}}_{*j}\left[\widetilde{\mathbf{C}}_{jj} + \sigma_j^2\mathbb{I}_j\right]^{-1}\widetilde{\mathbf{C}}_{j*}\right). \tag{14}$$

In order to obtain a sparse form of the predictive distribution we need to make an additional assumption.

**Assumption 2** *We assume that $\mathbf{f}_m$ is sufficient for $\bar{f}(\boldsymbol{x}^*)$, i.e.,*

$$\Pr(\bar{f}(\boldsymbol{x}^*)|\mathbf{f}_m, \breve{\boldsymbol{y}}) = \Pr(\bar{f}(\boldsymbol{x}^*)|\mathbf{f}_m),$$

*implying that*

$$\bar{f}(\boldsymbol{x}^*)|\mathbf{f}_m, \breve{\boldsymbol{y}} \sim \mathcal{N}\left(\mathbf{C}_{*m}\mathbf{C}_{mm}^{-1}\mathbf{f}_m, \mathbf{C}_{**} - \mathbf{C}_{*m}\mathbf{C}_{mm}^{-1}\mathbf{C}_{m*}\right). \tag{15}$$

The above set of conditional distributions also imply that $\bar{f}(\boldsymbol{x}^*)$ and $\tilde{f}^j(\boldsymbol{x}^*)$ are independent given $\mathbf{f}_m$ and $\breve{\boldsymbol{y}}$.

To evaluate (12), we have the following

- We can easily get $\Pr(\bar{f}(\boldsymbol{x}^*)|\breve{\boldsymbol{y}})$ by marginalizing out $\mathbf{f}_m|\breve{\boldsymbol{y}}$ in (15),

$$\Pr(\bar{f}(\boldsymbol{x}^*)|\breve{\boldsymbol{y}}) = \int \Pr(\bar{f}(\boldsymbol{x}^*)|\mathbf{f}_m)\phi^*(\mathbf{f}_m)d\mathbf{f}_m$$

  yielding

$$\bar{f}(\boldsymbol{x}^*)|\breve{\boldsymbol{y}} \sim \mathcal{N}\left(\mathbf{C}_{*m}\mathbf{C}_{mm}^{-1}\mu, \mathbf{C}_{**} - \mathbf{C}_{*m}\mathbf{C}_{mm}^{-1}\mathbf{C}_{m*} + \mathbf{C}_{*m}\mathbf{C}_{mm}^{-1}A\mathbf{C}_{mm}^{-1}\mathbf{C}_{m*}\right). \tag{16}$$

- Similarly, we can obtain $\Pr(\tilde{f}(\boldsymbol{x}^*)|\breve{\boldsymbol{y}})$ by first calculating $\Pr(\mathbf{f}^j|\breve{\boldsymbol{y}})$ by marginalizing out $\mathbf{f}_m|\breve{\boldsymbol{y}}$ in (13) and then marginalizing out $\mathbf{f}^j|\breve{\boldsymbol{y}}$ in (14), as follows. First we have $\mathbf{f}^j|\breve{\boldsymbol{y}} \sim \mathcal{N}(\mathbf{C}_{jm}\mathbf{C}_{mm}^{-1}\mu, \boldsymbol{B})$ where

$$\boldsymbol{B} = \mathbf{C}_{jj} - \mathbf{C}_{jm}\mathbf{C}_{mm}^{-1}\mathbf{C}_{mj} + \mathbf{C}_{jm}\mathbf{C}_{mm}^{-1}A\mathbf{C}_{mm}^{-1}\mathbf{C}_{mj}.$$

  Next for $\Pr(\tilde{f}(\boldsymbol{x}^*)|\breve{\boldsymbol{y}})$, we have

$$\Pr(\tilde{f}^j(\boldsymbol{x}^*)|\breve{\boldsymbol{y}}) = \int \Pr(\tilde{f}^j(\boldsymbol{x}^*)|\mathbf{f}^j, \boldsymbol{y}^j)\Pr(\mathbf{f}^j|\breve{\boldsymbol{y}})d\mathbf{f}^j$$

and marginalizing out $\mathbf{f}^j$, $\tilde{f}(\boldsymbol{x}^*)|\breve{\boldsymbol{y}}$ can be obtained as

$$\mathcal{N}\Bigg(\widetilde{\mathbf{C}}_{*j}\left[\widetilde{\mathbf{C}}_{jj}+\sigma_j^2\mathbb{I}_j\right]^{-1}\left(\boldsymbol{y}^j-\mathbf{C}_{jm}\mathbf{C}_{mm}^{-1}\mu\right),\widetilde{\mathbf{C}}_{**}-\widetilde{\mathbf{C}}_{*j}\left[\widetilde{\mathbf{C}}_{jj}+\sigma_j^2\mathbb{I}_j\right]^{-1}\widetilde{\mathbf{C}}_{j*} \\ +\widetilde{\mathbf{C}}_{*j}\left[\widetilde{\mathbf{C}}_{jj}+\sigma_j^2\mathbb{I}_j\right]^{-1}\mathbf{C}_{jm}\mathbf{C}_{mm}^{-1}\times B\mathbf{C}_{mm}^{-1}\mathbf{C}_{mj}\left(\widetilde{\mathbf{C}}_{jj}+\sigma_j^2\mathbb{I}_j\right)^{-1}\widetilde{\mathbf{C}}_{j*}\Bigg). \tag{17}$$

- Finally, to calculate $\mathbf{Cov}[\bar{f}(\boldsymbol{x}^*)\tilde{f}^j(\boldsymbol{x}^*)|\breve{\boldsymbol{y}}]$ we have

$$\mathbf{Cov}[\bar{f}(\boldsymbol{x}^*)\tilde{f}_j(\boldsymbol{x}^*)|\breve{\boldsymbol{y}}]=\mathbb{E}\left[\bar{f}^j(\boldsymbol{x}^*)\cdot\tilde{f}^j(\boldsymbol{x}^*)|\breve{\boldsymbol{y}}\right]-\mathbb{E}\left[\bar{f}(\boldsymbol{x}^*)|\breve{\boldsymbol{y}}\right]\mathbb{E}\left[\tilde{f}(\boldsymbol{x}^*)|\breve{\boldsymbol{y}}\right]$$

where

$$\begin{aligned}\mathbb{E}\left[\bar{f}^j(\boldsymbol{x}^*)\cdot\tilde{f}^j(\boldsymbol{x}^*)|\breve{\boldsymbol{y}}\right]&=\mathbb{E}_{\mathbf{f}_m|\breve{\boldsymbol{y}}}\mathbb{E}\left[\bar{f}^j(\boldsymbol{x}^*)\cdot\tilde{f}^j(\boldsymbol{x}^*)|\mathbf{f}_m,\breve{\boldsymbol{y}}\right]\\&=\mathbb{E}_{\mathbf{f}_m|\breve{\boldsymbol{y}}}\left[\mathbb{E}\left[\bar{f}^j(\boldsymbol{x}^*)|\mathbf{f}_m\right]\cdot\mathbb{E}\left[\tilde{f}^j(\boldsymbol{x}^*)|\mathbf{f}_m,\boldsymbol{y}^j\right]\right]\end{aligned} \tag{18}$$

where the second line holds because, as observed above, the terms are conditionally independent. The first term $\mathbb{E}\left[\bar{f}^j(\boldsymbol{x}^*)|\mathbf{f}_m\right]$ can be obtained directly from (15). By marginalizing out $\mathbf{f}^j|\mathbf{f}_m$ in (14) such that

$$\Pr(\tilde{f}^j(\boldsymbol{x}^*)|\mathbf{f}_m,\boldsymbol{y}^j)=\int\Pr(\tilde{f}^j(\boldsymbol{x}^*)|\mathbf{f}^j,\breve{\boldsymbol{y}})\Pr(\mathbf{f}^j|\mathbf{f}_m)d\mathbf{f}^j,$$

we can get the second term. This yields

$$\mathcal{N}\Bigg(\widetilde{\mathbf{C}}_{*j}\left[\widetilde{\mathbf{C}}_{jj}+\sigma_j^2\mathbb{I}_j\right]^{-1}\left(\boldsymbol{y}^j-\mathbf{C}_{jm}\mathbf{C}_{mm}^{-1}\mathbf{f}_m\right),\widetilde{\mathbf{C}}_{**}-\widetilde{\mathbf{C}}_{*j}\left[\widetilde{\mathbf{C}}_{jj}+\sigma_j^2\mathbb{I}_j\right]^{-1}\widetilde{\mathbf{C}}_{j*} \\ +\widetilde{\mathbf{C}}_{*j}\left[\widetilde{\mathbf{C}}_{jj}+\sigma_j^2\mathbb{I}_j\right]^{-1}\boldsymbol{C}\left(\widetilde{\mathbf{C}}_{jj}+\sigma_j^2\mathbb{I}_j\right)^{-1}\widetilde{\mathbf{C}}_{j*}\Bigg) \tag{19}$$

where $\boldsymbol{C}=\mathbf{C}_{jj}-\mathbf{C}_{jm}\mathbf{C}_{mm}^{-1}\mathbf{C}_{mj}$. To simplify the notation, let $\mathbf{H}=\mathbf{C}_{*m}\mathbf{C}_{mm}^{-1}$, $\mathbf{F}=\widetilde{\mathbf{C}}_{*j}\left(\widetilde{\mathbf{C}}_{jj}+\sigma_j^2\mathbb{I}_j\right)^{-1}$ and $\mathbf{G}=\mathbf{C}_{jm}\mathbf{C}_{mm}^{-1}$. Then (18) can be evaluated as

$$\mathbf{H}\boldsymbol{y}^j\mathbf{F}\cdot\mathbb{E}[\mathbf{f}_m]-\mathbf{FG}\left(\mathbb{E}\left[\mathbf{f}_m\mathbf{f}_m^{\mathrm{T}}|\breve{\boldsymbol{y}}\right]\right)\mathbf{H}^{\mathrm{T}}=\mathbf{H}\boldsymbol{y}^j\mathbf{F}\cdot\mu-\mathbf{FG}\left[\boldsymbol{A}+\boldsymbol{\mu}\boldsymbol{\mu}^{\mathrm{T}}\right]\mathbf{H}^{\mathrm{T}}.$$

We have therefore shown how to calculate the predictive distribution in (12). The complexity of these computations is $\mathcal{O}(N_j^3+m^3)$ which is a significant improvement over $\mathcal{O}(N^3)$ where $N=M\times N_j$.

## 4. Sparse Grouped mixed-effect GP Model (GMT-GP)

In this section, we extend the mixed-effect GP model such that the fixed-effect functions admit a group structure. We call this *Grouped mixed-effect GP model* (GMT-GP). More precisely, each task is sampled from a mixture of shared fixed-effect GPs and then adds its individual variation. We show how to perform the inference and model selection efficiently.

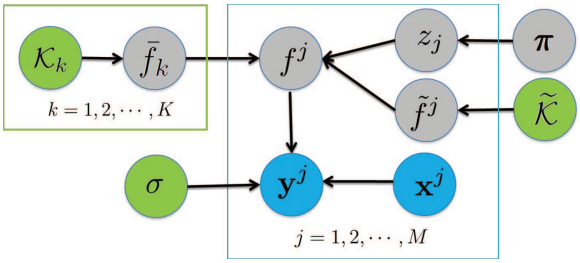


Figure 1: Plate graph of the GMT-GP. Blue nodes denote observations, green ones are (hyper)parameters and the gray ones are latent variables.

### 4.1. Generative Model

First, we specify the sparse GMT-GP model, and show how we can learn the hyper-parameters and the inducing variables using this sparse model.

**Assumption 3** *For each $j$ and $\boldsymbol{x} \in \mathcal{X}$,*

$$f^j(\boldsymbol{x}) = \bar{f}_{z_j}(\boldsymbol{x}) + \tilde{f}^j(\boldsymbol{x}), \quad j = 1, \cdots, M$$

*where $\{\bar{f}_k\}, k = 1, \cdots, K$ and $\tilde{f}^j$ are zero-mean Gaussian processes with covariance function $\mathcal{K}_k$ and $\widetilde{\mathcal{K}}$, and $z_j \in \{1, \cdots, K\}$. In addition, $\{\bar{f}_k\}$ and $\{\tilde{f}^j\}$ are assumed to be mutually independent.*

The generative process (shown in Fig. 1) is as follows, where **Dir** and **Multi** denote the Dirichlet and the Multinominal distribution respectively.

1. Draw the processes of the mean effect: $\bar{f}_k(\cdot)|\boldsymbol{\theta}_k \sim \mathcal{GP}(0, \mathcal{K}_k(\cdot,\cdot)), \quad k = 1, 2, \cdots, K$;
2. Draw $\boldsymbol{\pi}|\boldsymbol{\alpha}_0 \sim \mathbf{Dir}(\boldsymbol{\alpha}_0)$;
3. For the $j$-th task (time series);
   - Draw $z_j|\boldsymbol{\pi} \sim \mathbf{Multi}(\boldsymbol{\pi})$;
   - Draw the random effect: $\tilde{f}^j(\cdot)|\tilde{\boldsymbol{\theta}} \sim \mathcal{GP}(0, \widetilde{\mathcal{K}}(\cdot,\cdot))$;
   - Draw $\boldsymbol{y}^j|z_j, f^j, \boldsymbol{x}^j, \sigma_j^2 \sim \mathcal{N}\left(f^j(\boldsymbol{x}^j), \sigma_j^2 \cdot \mathbb{I}_j\right)$, where $f^j = \bar{f}_{z_j} + \tilde{f}^j$ and where to simplify the notation $\mathbb{I}_j$ stands for $\mathbb{I}_{N_j}$.

### 4.2. Variational Model Selection

In this section we show how to perform the learning via variational approximation. The derivation follows the same outline as in the previous section but due to the hidden variables $z_j$ that specify group membership, we have to use the variational EM algorithm. As mentioned above, for the $k$-th mixed-effect (or center), we introduce $m_k$ auxiliary inducing support variables $\mathcal{X}_m^k$ and the hidden variable $\boldsymbol{\eta}_k = \bar{f}_k(\mathcal{X}_m^k)$, which is the value of $k$-th fixed-effect function evaluated at $\mathcal{X}_m^k$.

Let $\mathfrak{f}_k = \bar{f}_k(\breve{\boldsymbol{x}}) \in \mathbb{R}^N$ denote the function values of the $k$-th mean effect so that $\mathbf{f}_k^j = \bar{f}_k(\boldsymbol{x}^j) \in \mathbb{R}^{N_j}$ is the sub-vector of $\mathfrak{f}_k$ corresponding to the $j$-th task. Let $\tilde{\mathbf{f}}^j = \tilde{f}(\boldsymbol{x}^j) \in \mathbb{R}^{N_j}$ be the values of the random effect at $\boldsymbol{x}^j$. Denote the collection of the hidden variables as $\mathfrak{F} = \{\mathfrak{f}_k\}, \widetilde{\mathcal{F}} = \{\tilde{\mathbf{f}}^j\}, \mathbf{H} = \{\boldsymbol{\eta}_k\}$, $\mathbf{Z} = \{z_j\}$, and $\boldsymbol{\pi}$. In addition let $\mathbf{C}^k_{*j} = \mathcal{K}_k(\boldsymbol{x}^*, \boldsymbol{x}^j)$, $\mathbf{C}^k_{jj} = \mathcal{K}_k(\boldsymbol{x}^j, \boldsymbol{x}^j), \mathbf{C}_{jk} = \mathcal{K}_k(\boldsymbol{x}^j, \mathcal{X}^k_m)$ and $\mathbf{C}_{kk} = \mathcal{K}_k(\mathcal{X}^k_m, \mathcal{X}^k_m)$, and similarly $\widetilde{\mathbf{C}}_{*j} = \widetilde{\mathcal{K}}(\boldsymbol{x}^*, \boldsymbol{x}^j), \widetilde{\mathbf{C}}_{jj} = \widetilde{\mathcal{K}}(\boldsymbol{x}^j, \boldsymbol{x}^j)$ and $\widehat{\mathbf{C}}_{jj} = \widetilde{\mathbf{C}}_{jj} + \sigma_j^2 \mathbb{I}_j$ where $\mathbb{I}_j$ stands for $\mathbb{I}_{N_j}$.

To learn the hyper-parameters we wish to maximize the marginal likelihood $\Pr(\breve{\boldsymbol{y}})$ where $\breve{\boldsymbol{y}}$ is all the measurements. In the following we develop a variational lower bound for this quantity. To this end, we need the complete data likelihood and the variational distribution. The complete data likelihood is given by

$$\Pr(\breve{\boldsymbol{y}}, \mathfrak{F}, \widetilde{\mathcal{F}}, \mathbf{H}, \mathbf{Z}, \boldsymbol{\pi}) = \Pr(\breve{\boldsymbol{y}}|\mathfrak{F}, \widetilde{\mathcal{F}}, \mathbf{Z}) \Pr(\mathfrak{F}|\mathbf{H}) \Pr(\mathbf{Z}|\boldsymbol{\pi}) \Pr(\boldsymbol{\pi}) \Pr(\widetilde{\mathcal{F}}) \Pr(\mathbf{H}) \tag{20}$$

where

$$\Pr(\mathbf{H}) = \prod_{k=1}^{K} \Pr(\boldsymbol{\eta}_k), \qquad \Pr(\widetilde{\mathcal{F}}) = \prod_{j=1}^{M} \Pr(\tilde{\mathbf{f}}^j),$$

$$\Pr(\boldsymbol{\pi}) = \mathbf{Dir}(\boldsymbol{\pi}|\alpha_0), \qquad \Pr(\mathbf{Z}|\boldsymbol{\pi}) = \prod_{j=1}^{M}\prod_{k=1}^{K} \pi_k^{z_{jk}}$$

$$\Pr(\mathfrak{F}|\mathbf{H}) = \prod_{k=1}^{K} \Pr(\mathfrak{f}_k|\boldsymbol{\eta}_k), \quad \Pr(\breve{\boldsymbol{y}}|\mathfrak{F}, \widetilde{\mathcal{F}}, \mathbf{Z}) = \prod_{j=1}^{M}\prod_{k=1}^{K} \Big[\Pr(\boldsymbol{y}^j|\tilde{\mathbf{f}}^j, \mathfrak{f}_k)\Big]^{z_{jk}}$$

where, as usual $\{z_{jk}\}$ represent $z_j$ as a unit vector.

We approximate the posterior $\Pr(\mathfrak{F}, \widetilde{\mathcal{F}}, \mathbf{H}, \mathbf{Z}, \boldsymbol{\pi}|\breve{\boldsymbol{y}})$ on the hidden variables using

$$q(\mathfrak{F}, \widetilde{\mathcal{F}}, \mathbf{H}, \mathbf{Z}, \boldsymbol{\pi}) = q(\mathfrak{F}, \widetilde{\mathcal{F}}, \mathbf{H}|\mathbf{Z}) q(\mathbf{Z}) q(\boldsymbol{\pi}) \tag{21}$$

where

$$\begin{aligned} q(\mathfrak{F}, \widetilde{\mathcal{F}}, \mathbf{H}|\mathbf{Z}) &= \Pr(\widetilde{\mathcal{F}}|\mathfrak{F}, \mathbf{Z}, \breve{\boldsymbol{y}}) \Pr(\mathfrak{F}|\mathbf{H}) \boldsymbol{\Phi}(\mathbf{H}) \\ &= \prod_{j=1}^{M}\prod_{k=1}^{K} \Big[\Pr(\tilde{\mathbf{f}}^j|\mathfrak{f}_k, \boldsymbol{y}^j)\Big]^{z_{jk}} \prod_{k=1}^{K} \Pr(\mathfrak{f}_k|\boldsymbol{\eta}_k) \phi(\boldsymbol{\eta}_k). \end{aligned}$$

This extends the variation form of the previous section. Our use of $\mathfrak{f}_k$ as the complete set of observation when the true group is $k$ makes for convenient notation of simplifying the derivation.

The variational lower bound, denoted as $F_V$, is given by:

$$\begin{aligned}
\Pr(\breve{\boldsymbol{y}}) \geqslant F_V &= \int q(\mathfrak{F}, \widetilde{\mathcal{F}}, \mathbf{H}, \mathbf{Z}, \boldsymbol{\pi}) \times \log \left[ \frac{\Pr(\breve{\boldsymbol{y}}, \mathfrak{F}, \widetilde{\mathcal{F}}, \mathbf{H}, \mathbf{Z}, \boldsymbol{\pi})}{q(\mathfrak{F}, \widetilde{\mathcal{F}}, \mathbf{H}, \mathbf{Z}, \boldsymbol{\pi})} \right] d\mathfrak{F}\, d\widetilde{\mathcal{F}}\, d\mathbf{H}\, d\mathbf{Z}\, d\boldsymbol{\pi} \\
&= \int q(\boldsymbol{\pi}) q(\mathbf{Z}) q(\mathfrak{F}, \widetilde{\mathcal{F}}, \mathbf{H}|\mathbf{Z}) \\
&\quad \times \log \left[ \frac{\Pr(\breve{\boldsymbol{y}}|\mathfrak{F}, \widetilde{\mathcal{F}}, \mathbf{Z}) \Pr(\mathfrak{F}|\mathbf{H}) \Pr(\mathbf{Z}|\boldsymbol{\pi}) \Pr(\pi) \Pr(\widetilde{\mathcal{F}}) \Pr(\mathbf{H})}{q(\mathfrak{F}, \widetilde{\mathcal{F}}, \mathbf{H}|\mathbf{Z}) q(\mathbf{Z}) q(\boldsymbol{\pi})} \right] d\mathfrak{F} d\widetilde{\mathcal{F}} d\mathbf{H} d\mathbf{Z} d\boldsymbol{\pi} \\
&= \int q(\mathbf{Z}) q(\boldsymbol{\pi}) \log \left[ \frac{\Pr(\boldsymbol{\pi}) \Pr(\mathbf{Z}|\boldsymbol{\pi})}{q(\mathbf{Z}) q(\boldsymbol{\pi})} \right] d\boldsymbol{\pi} d\mathbf{Z} \\
&\quad + \int q(\mathbf{Z}) q(\mathfrak{F}, \widetilde{\mathcal{F}}, \mathbf{H}|\mathbf{Z}) \log \left[ \frac{\Pr(\breve{\boldsymbol{y}}|\mathfrak{F}, \widetilde{\mathcal{F}}, \mathbf{Z}) \Pr(\mathfrak{F}|\mathbf{H}) \Pr(\widetilde{\mathcal{F}}) \Pr(\mathbf{H})}{q(\mathfrak{F}, \widetilde{\mathcal{F}}, \mathbf{H}|\mathbf{Z})} \right] d\mathfrak{F} d\widetilde{\mathcal{F}} d\mathbf{H} d\mathbf{Z}
\end{aligned}$$

To begin with, we evaluate the second term denoted as $F_{V2}$, as follows. The term inside the log can be evaluated as

$$\begin{aligned}
\boldsymbol{\Delta} &= \frac{\Pr(\breve{\boldsymbol{y}}|\mathfrak{F}, \widetilde{\mathcal{F}}, \mathbf{Z}) \Pr(\mathfrak{F}|\mathbf{H}) \Pr(\widetilde{\mathcal{F}}) \Pr(\mathbf{H})}{q(\mathfrak{F}, \widetilde{\mathcal{F}}, \mathbf{H}|\mathbf{Z})} \\
&= \frac{\prod_{j,k} \left[ \Pr(\boldsymbol{y}^j|\tilde{\mathbf{f}}^j, \mathfrak{f}_k) \right]^{z_{jk}} \prod_k \Pr(\mathfrak{f}_k|\boldsymbol{\eta}_k) \prod_j \Pr(\tilde{\mathbf{f}}^j) \prod_k \Pr(\boldsymbol{\eta}_k)}{\prod_{j,k} \left[ \Pr(\tilde{\mathbf{f}}^j|\mathfrak{f}_k, \boldsymbol{y}^j) \right]^{z_{jk}} \prod_k \Pr(\mathfrak{f}_k|\boldsymbol{\eta}_k) \phi(\boldsymbol{\eta}_k)} \\
&= \prod_{j=1}^{m} \prod_{k=1}^{K} \left[ \frac{\Pr(\boldsymbol{y}^j|\mathfrak{f}_k, \tilde{\mathbf{f}}^j) \Pr(\tilde{\mathbf{f}}^j)}{\Pr(\tilde{\mathbf{f}}^j|\mathfrak{f}_k, \boldsymbol{y}^j)} \right]^{z_{jk}} \times \prod_{k=1}^{K} \frac{\Pr(\boldsymbol{\eta}_k)}{\phi(\boldsymbol{\eta}_k)}.
\end{aligned}$$

Thus, we can write $F_{V2}$ as

$$\begin{aligned}
F_{V2} &= \int q(\mathbf{Z}) q(\mathfrak{F}, \widetilde{\mathcal{F}}, \mathbf{H}|\mathbf{Z}) (\log \boldsymbol{\Delta})\; d\mathfrak{F} d\widetilde{\mathcal{F}} d\mathbf{H} d\mathbf{Z} \\
&= \int q(\mathbf{Z}) \left[ \int \prod_{k=1}^{K} \phi(\boldsymbol{\eta}_k) \left\{ \log G(\mathbf{Z}, \mathbf{H}, \breve{\boldsymbol{y}}) + \sum_{k=1}^{K} \log \left[ \frac{\Pr(\boldsymbol{\eta}_k)}{\phi(\boldsymbol{\eta}_k)} \right] \right\} d\mathbf{H} \right] d\mathbf{Z},
\end{aligned}$$

where

$$\log G(\mathbf{Z}, \mathbf{H}, \breve{\boldsymbol{y}}) = \int \Pr(\widetilde{\mathcal{F}}|\mathfrak{F}, \mathbf{Z}) \Pr(\mathfrak{F}|\mathbf{H}) \log \left[ \prod_{j=1}^{M} \prod_{k=1}^{K} \left[ \frac{\Pr(\boldsymbol{y}^j|\mathfrak{f}_k, \tilde{\mathbf{f}}^j) \Pr(\tilde{\mathbf{f}}^j)}{\Pr(\tilde{\mathbf{f}}^j|\mathfrak{f}_k, \boldsymbol{y}^j)} \right]^{z_{jk}} \right] d\mathfrak{F} d\widetilde{\mathcal{F}}.$$

We show below that $\log G(\mathbf{Z}, \mathbf{H}, \breve{\boldsymbol{y}})$ can be decomposed as

$$\log G(\mathbf{Z}, \mathbf{H}, \breve{\boldsymbol{y}}) = \sum_{j=1}^{M} \sum_{k=1}^{K} z_{jk} \log G(\boldsymbol{\eta}_k, \boldsymbol{y}^j),$$

where

$$\log G(\boldsymbol{\eta}_k, \boldsymbol{y}^j) = \log \left[\mathcal{N}(\boldsymbol{y}^j | \boldsymbol{\alpha}_j^k, \widehat{\mathbf{C}}_{jj})\right] - \frac{1}{2}\mathbf{Tr}\left[(\mathbf{C}_{jj}^k - \boldsymbol{Q}_{jj}^k)\widehat{\mathbf{C}}_{jj}^{-1}\right], \tag{22}$$

where $\boldsymbol{\alpha}_j^k = \mathbf{C}_{jk}\mathbf{C}_{kk}^{-1}\boldsymbol{\eta}_k$ and $\boldsymbol{Q}_{jj}^k = \mathbf{C}_{jk}\mathbf{C}_{kk}^{-1}\mathbf{C}_{kj}$. Consequently, the variational lower bound is

$$\begin{aligned} F_V = &\int q(\mathbf{Z})q(\pi)\log\left[\frac{\Pr(\boldsymbol{\pi})\Pr(\mathbf{Z}|\boldsymbol{\pi})}{q(\mathbf{Z})q(\boldsymbol{\pi})}\right]d\boldsymbol{\pi}d\mathbf{Z} \\ &+ \int q(\mathbf{Z})\left[\int \prod_{k=1}^{K}\phi(\boldsymbol{\eta}_k)\left\{\log G(\mathbf{Z}, \mathbf{H}, \breve{\boldsymbol{y}}) + \sum_{k=1}^{K}\log\left[\frac{\Pr(\boldsymbol{\eta}_k)}{\phi(\boldsymbol{\eta}_k)}\right]\right\}d\mathbf{H}\right]d\mathbf{Z} \end{aligned}$$

To optimize the parameters we use the variational EM algorithm.

- In the **Variational E-Step**, we estimate $q^*(\mathbf{Z}), q^*(\boldsymbol{\pi})$ and $\{\phi^*(\boldsymbol{\eta}_k)\}$.

  To get the variational distribution $q^*(\mathbf{Z})$, we take derivative of $F_V$ w.r.t. $q(\mathbf{Z})$ and set it to 0. This yields

$$\begin{aligned} \log q^*(\mathbf{Z}) &= \int q(\boldsymbol{\pi})\log(\Pr(\mathbf{Z}|\boldsymbol{\pi}))d\boldsymbol{\pi} + \int \prod_{k=1}^{K}\phi(\boldsymbol{\eta}_k)\log G(\mathbf{Z}, \mathbf{H}, \breve{\boldsymbol{y}})d\mathbf{H} \\ &= \sum_{j=1}^{M}\sum_{k=1}^{K} z_{jk}\left[\mathbb{E}_{q(\boldsymbol{\pi})}[\log \pi_k] + \mathbb{E}_{\phi(\boldsymbol{\eta}_k)}[\log G(\boldsymbol{\eta}_k, \boldsymbol{y}^j)]\right] + \mathbf{const} \end{aligned}$$

  from which we can derive

$$\begin{aligned} q^*(\mathbf{Z}) &= \prod_{j=1}^{M}\prod_{k=1}^{K} r_{jk}^{z_{jk}}, \qquad r_{jk} = \frac{\rho_{jk}}{\sum_{k=1}^{K}\rho_{jk}} \\ \log \rho_{jk} &= \mathbb{E}_{q(\boldsymbol{\pi})}[\log \pi_k] + \mathbb{E}_{\phi(\boldsymbol{\eta}_k)}[\log G(\boldsymbol{\eta}_k, \boldsymbol{y}^j)], \end{aligned}$$

  where $\mathbb{E}_{q(\boldsymbol{\pi})}[\log \pi_k] = \psi(\alpha_k) - \psi(\sum_k \alpha_k)$ where $\psi$ is the digamma function, $\alpha_k$ is defined below in (23), and $\mathbb{E}_{\phi(\boldsymbol{\eta}_k)}[\log G(\boldsymbol{\eta}_k, \boldsymbol{y}^j)]$ is given below in (34).

  For the variational distribution of $q^*(\boldsymbol{\pi})$ the derivative yields

$$\begin{aligned} \log q^*(\boldsymbol{\pi}) &= \log \Pr(\boldsymbol{\pi}) + \int q(\mathbf{Z})\log(\Pr(\mathbf{Z}|\boldsymbol{\pi}))d\boldsymbol{\pi} + \mathbf{const} \\ &= (\alpha_0 - 1)\sum_{k=1}^{K}\log(\pi_k) + \sum_{j=1}^{M}\sum_{k=1}^{K}\mathbb{E}[z_{jk}]\log \pi_k + \mathbf{const} \end{aligned}$$

  and taking the exponential of both sides, we have

$$q^*(\boldsymbol{\pi}) = \mathbf{Dir}(\boldsymbol{\pi}|\boldsymbol{\alpha})$$

where

$$\alpha_k = \alpha_0 + N_k, \quad N_k = \sum_{j=1}^{K} r_{jk}. \tag{23}$$

The final step is to get the variational distribution of $\phi^*(\boldsymbol{\eta}_k), k = 1, \cdots, K$. Notice that only $F_{V2}$ is a function of $\phi(\boldsymbol{\eta}_k)$. We can rewrite this portion as

$$\begin{aligned}
&\int \prod_{k=1}^{K} \phi(\boldsymbol{\eta}_k) \left( \left\{ \int q(\mathbf{Z}) \sum_{j=1}^{M} \sum_{k=1}^{K} z_{jk} \left[ \log G(\boldsymbol{\eta}_k, \boldsymbol{y}^j) \right] d\mathbf{Z} \right\} + \sum_{k=1}^{K} \log \left[ \frac{\Pr(\boldsymbol{\eta}_k)}{\phi(\boldsymbol{\eta}_k)} \right] \right) d\mathbf{H} \\
&= \int \prod_{k=1}^{K} \phi(\boldsymbol{\eta}_k) \left( \sum_{j=1}^{M} \sum_{k=1}^{K} \mathbb{E}_{q(\mathbf{Z})}[z_{jk}] \left[ \log G(\boldsymbol{\eta}_k, \boldsymbol{y}^j) \right] + \sum_{k=1}^{K} \log \left[ \frac{\Pr(\boldsymbol{\eta}_k)}{\phi(\boldsymbol{\eta}_k)} \right] \right) d\mathbf{H} \\
&= \sum_{k=1}^{K} \int \phi(\boldsymbol{\eta}_k) \left\{ \left[ \sum_{j=1}^{M} \mathbb{E}_{q(\mathbf{Z})}[z_{jk}] \log G(\boldsymbol{\eta}_k, \boldsymbol{y}^j) \right] + \log \left[ \frac{\Pr(\boldsymbol{\eta}_k)}{\phi(\boldsymbol{\eta}_k)} \right] \right\} d\boldsymbol{\eta}_k.
\end{aligned} \tag{24}$$

Thus, our task reduces to find $\phi^*(\boldsymbol{\eta}_k)$ separately. Taking the derivative of (24) w.r.t. $\phi(\boldsymbol{\eta}_k)$ and setting it to be zero, we have

$$\log \phi^*(\boldsymbol{\eta}_k) = \sum_{j=1}^{M} \mathbb{E}_{q(\mathbf{Z})}[z_{jk}] \log G(\boldsymbol{\eta}_k, \boldsymbol{y}^j) + \log \Pr(\boldsymbol{\eta}_k) + \textbf{const.}$$

Using (22) and the fact that second term in (22) is not a function of $\boldsymbol{\eta}_k$, we obtain

$$\phi^*(\boldsymbol{\eta}_k) \propto \prod_{j=1}^{M} \left[ \mathcal{N}(\boldsymbol{y}^j | \boldsymbol{\alpha}_j^k, \widehat{\mathbf{C}}_{jj}^k) \right]^{\mathbb{E}_{q(\mathbf{Z})}[z_{jk}]} \Pr(\boldsymbol{\eta}_k). \tag{25}$$

Thus, we have

$$\phi^*(\boldsymbol{\eta}_k) \propto \exp \left\{ -\frac{1}{2} (\boldsymbol{\eta}^k)^T \left( \mathbf{C}_{kk}^{-1} \boldsymbol{\Phi} \mathbf{C}_{kk}^{-1} \right) \boldsymbol{\eta}^k + (\boldsymbol{\eta}_k)^T \left( \mathbf{C}_{kk}^{-1} \sum_{j=1}^{M} \mathbb{E}_{q(\mathbf{Z})}[z_{jk}] \mathbf{C}_{kj} [\widehat{\mathbf{C}}_{jj}]^{-1} \boldsymbol{y}_j \right) \right\},$$

where

$$\boldsymbol{\Phi} = \mathbf{C}_{kk} + \sum_{j=1}^{M} \mathbb{E}_{q(\mathbf{Z})}[z_{jk}] \mathbf{C}_{kj} [\widehat{\mathbf{C}}_{jj}]^{-1} \mathbf{C}_{jk}.$$

Completing the square yields the Gaussian distribution

$$\phi^*(\boldsymbol{\eta}_k) = \mathcal{N}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k),$$

where

$$\boldsymbol{\mu}_k = \mathbf{C}_{kk} \boldsymbol{\Phi}^{-1} \sum_{j=1}^{M} \mathbb{E}_{q(\mathbf{Z})}[z_{jk}] \mathbf{C}_{kj} [\widehat{\mathbf{C}}_{jj}]^{-1} \boldsymbol{y}_j, \quad \boldsymbol{\Sigma}_k = \mathbf{C}_{kk} \boldsymbol{\Phi}^{-1} \mathbf{C}_{kk}. \tag{26}$$

- In the **Variational M-Step**, based on the previous estimated variational distribution, we wish to find hyperparameters that maximize the variational lower bound $F_V$. The terms that depend on the hyperparameters and the inducing variables $\{\mathcal{X}_m^k\}$ are given in (24). Therefore, using (22) again, we have

$$\begin{aligned}
F_V(\mathcal{X}_k, \boldsymbol{\theta}) &= \sum_{k=1}^{K} \int \phi^*(\boldsymbol{\eta}_k) \left\{ \left[ \sum_{j=1}^{M} r_{jk} \log G(\boldsymbol{\eta}_k, \boldsymbol{y}^j) \right] + \log \left[ \frac{\Pr(\boldsymbol{\eta}_k)}{\phi^*(\boldsymbol{\eta}_k)} \right] \right\} d\boldsymbol{\eta}_k \\
&= \sum_{k=1}^{K} \int \phi^*(\boldsymbol{\eta}_k) \left\{ \log \left[ \sum_{j=1}^{M} r_{jk} \mathcal{N}(\boldsymbol{y}^j | \boldsymbol{\alpha}_j^k, \widehat{\mathbf{C}}_{jj}) \right] + \log \left[ \frac{\Pr(\boldsymbol{\eta}_k)}{\phi^*(\boldsymbol{\eta}_k)} \right] \right\} d\boldsymbol{\eta}_k \\
&\qquad - \frac{1}{2} \sum_{k=1}^{K} \sum_{j=1}^{m} r_{jk} \mathbf{Tr} \left[ (\mathbf{C}_{jj}^k - \boldsymbol{Q}_{jj}^k) \widehat{\mathbf{C}}_{jj}^{-1} \right] \\
&= \sum_{k=1}^{K} \int \phi^*(\boldsymbol{\eta}_k) \left\{ \log \left[ \frac{\prod_{j=1}^{M} \left[ \mathcal{N}(\boldsymbol{y}^j | \boldsymbol{\alpha}_j^k, \widehat{\mathbf{C}}_{jj}) \right]^{r_{jk}} \Pr(\boldsymbol{\eta}_k)}{\phi^*(\boldsymbol{\eta}_k)} \right] \right\} d\boldsymbol{\eta}_k \\
&\qquad - \frac{1}{2} \sum_{k=1}^{K} \sum_{j=1}^{m} r_{jk} \mathbf{Tr} \left[ (\mathbf{C}_{jj}^k - \boldsymbol{Q}_{jj}^k) \widehat{\mathbf{C}}_{jj}^{-1} \right]
\end{aligned}$$

From (25), we know that the term inside the log is constant, and therefore, extracting the log from the integral and cancelling the $\phi*$ terms we see that the $k$'th element of first term is equal to the logarithm of

$$\int \prod_{j=1}^{M} \left[ \mathcal{N}(\boldsymbol{y}^j | \boldsymbol{\alpha}_j^k, \widehat{\mathbf{C}}_{jj}) \right]^{r_{jk}} \Pr(\boldsymbol{\eta}_k) d\boldsymbol{\eta}_k. \tag{27}$$

We next show how this multivariate integral can be evaluated. First consider

$$\begin{aligned}
&\left[ \mathcal{N}(\boldsymbol{y}^j | \boldsymbol{\alpha}_j^k, \widehat{\mathbf{C}}_{jj}) \right]^{r_{jk}} \\
&= \left( (2\pi)^{-\frac{N_j}{2}} |\widehat{\mathbf{C}}_{jj}|^{-\frac{1}{2}} \right)^{r_{jk}} \exp \left\{ -\frac{r_{jk}}{2} (\boldsymbol{y}^j - \boldsymbol{\alpha}_j^k)^T [\widehat{\mathbf{C}}_{jj}]^{-1} (\boldsymbol{y}^j - \boldsymbol{\alpha}_j^k) \right\} \\
&= \left( (2\pi)^{-\frac{N_j}{2}} |\widehat{\mathbf{C}}_{jj}|^{-\frac{1}{2}} \right)^{r_{jk}} \exp \left\{ -\frac{1}{2} (\boldsymbol{y}^j - \boldsymbol{\alpha}_j^k)^T \left[ r_{jk}^{-1} \widehat{\mathbf{C}}_{jj} \right]^{-1} (\boldsymbol{y}^j - \boldsymbol{\alpha}_j^k) \right\} \\
&= \frac{\left( (2\pi)^{-\frac{N_j}{2}} |\widehat{\mathbf{C}}_{jj}|^{-\frac{1}{2}} \right)^{r_{jk}}}{(2\pi)^{-\frac{N_j}{2}} |r_{jk}^{-1} \widehat{\mathbf{C}}_{jj}|^{-\frac{1}{2}}} \cdot (2\pi)^{-\frac{N_j}{2}} |r_{jk}^{-1} \widehat{\mathbf{C}}_{jj}|^{-\frac{1}{2}} \exp \left\{ -\frac{1}{2} (\boldsymbol{y}^j - \boldsymbol{\alpha}_j^k)^T \left[ r_{jk}^{-1} \widehat{\mathbf{C}}_{jj} \right]^{-1} (\boldsymbol{y}^j - \boldsymbol{\alpha}_j^k) \right\} \\
&= A_{jk} \mathcal{N}(\boldsymbol{y}^j | \boldsymbol{\alpha}_j^k, r_{jk}^{-1} \widehat{\mathbf{C}}_{jj}^k),
\end{aligned}$$

where $A_{jk} = (r_{jk})^{\frac{N_j}{2}} (2\pi)^{\frac{N_j(1-r_{jk})}{2}} |\widehat{\mathbf{C}}_{jj}|^{\frac{1-r_{jk}}{2}}$. Thus, we have

$$\prod_{j=1}^{M} \left[ \mathcal{N}(\boldsymbol{y}^j | \boldsymbol{\alpha}_j^k, \widehat{\mathbf{C}}_{jj}) \right]^{r_{jk}} = \left[ \prod_{j=1}^{M} A_{jk} \right] \prod_{j=1}^{M} \mathcal{N}(\boldsymbol{y}^j | \boldsymbol{\alpha}_j^k, r_{jk}^{-1} \widehat{\mathbf{C}}_{jj}).$$

The first part is not a function of $\boldsymbol{\eta}_k$, for the integration we are only interested in the second part. Since $\breve{\boldsymbol{y}}$ is the concatenation of all $\boldsymbol{y}^j$'s, we can write

$$\prod_{j=1}^{M} \mathcal{N}(\boldsymbol{y}^j | \boldsymbol{\alpha}_j^k, r_{jk}^{-1} \widehat{\mathbf{C}}_{jj}) = \mathcal{N}(\breve{\boldsymbol{y}} | \boldsymbol{\Lambda}_k \mathbf{C}_{kk}^{-1} \boldsymbol{\eta}_k, \widehat{\mathbf{C}}^k), \tag{28}$$

where

$$\boldsymbol{\Lambda}_k = \begin{pmatrix} \mathbf{C}_{1k} \\ \mathbf{C}_{2k} \\ \vdots \\ \mathbf{C}_{Mk} \end{pmatrix} \quad \text{and} \quad \widehat{\mathbf{C}}^k = \bigoplus_{j=1}^{M} r_{jk}^{-1} \widehat{\mathbf{C}}_{jj}^k = \begin{pmatrix} r_{1k}^{-1}\widehat{\mathbf{C}}_{11} & & & \\ & r_{2k}^{-1}\widehat{\mathbf{C}}_{22} & & \\ & & \ddots & \\ & & & r_{Mk}^{-1}\widehat{\mathbf{C}}_{MM} \end{pmatrix}.$$

Therefore, the integral can the written as the following marginal distribution of $\Pr(\breve{\boldsymbol{y}}|k)$,

$$\int \prod_{j=1}^{M} \mathcal{N}(\boldsymbol{y}^j | \boldsymbol{\alpha}_j^k, r_{jk}^{-1} \widehat{\mathbf{C}}_{jj}) \Pr(\boldsymbol{\eta}_k) d\boldsymbol{\eta}_k = \int \mathcal{N}(\breve{\boldsymbol{y}} | \boldsymbol{\Lambda}_k \mathbf{C}_{kk}^{-1} \boldsymbol{\eta}_k, \widehat{\mathbf{C}}^k) \Pr(\boldsymbol{\eta}_k) d\boldsymbol{\eta}_k = \Pr(\breve{\boldsymbol{y}}|k). \tag{29}$$

Using the fact that $\Pr(\boldsymbol{\eta}_k) = \mathcal{N}(\mathbf{0}, \mathbf{C}_{kk})$ and observing that (28) is a conditional Gaussian, we have

$$\Pr(\breve{\boldsymbol{y}}|k) = \mathcal{N}(\mathbf{0}, \boldsymbol{\Lambda}_k \mathbf{C}_{kk}^{-1} \boldsymbol{\Lambda}_k^T + \widehat{\mathbf{C}}^k).$$

Using this form and the portion of $A_{jk}$ that depends on the parameters we get

$$\begin{aligned} F_V(\mathcal{X}, \boldsymbol{\theta}) &= \sum_{k=1}^{K} \log \Pr(\breve{\boldsymbol{y}}|k) + \sum_{k=1}^{K}\sum_{j=1}^{M} \frac{1 - r_{jk}}{2} \log |\widehat{\mathbf{C}}_{jj}| - \frac{1}{2} \sum_{k=1}^{K}\sum_{j=1}^{M} r_{jk} \mathbf{Tr}\left[ (\mathbf{C}_{jj}^k - \boldsymbol{Q}_{jj}^k) \widehat{\mathbf{C}}_{jj}^{-1} \right] \\ &= \sum_{k=1}^{K} \log \Pr(\breve{\boldsymbol{y}}|k) + \frac{K-1}{2} \sum_{j=1}^{M} \log |\widehat{\mathbf{C}}_{jj}| - \frac{1}{2} \sum_{k=1}^{K}\sum_{j=1}^{M} r_{jk} \mathbf{Tr}\left[ (\mathbf{C}_{jj}^k - \boldsymbol{Q}_{jj}^k) \widehat{\mathbf{C}}_{jj}^{-1} \right] \end{aligned} \tag{30}$$

This extends the bound for the single center $K = 1$ case given in (10). Furthermore, following the same line as the previous derivation, the direct inference for the full model can be obtained where $\eta_k$ is substituted with $\mathfrak{f}_k$ and the variational lower bound becomes

$$F_V(\mathcal{X}, \boldsymbol{\theta}) = \sum_{k=1}^{K} \log \mathcal{N}(\breve{y} | \mathbf{0}, \mathbf{C}_{kk} + \widehat{\mathbf{C}}^k) + \frac{K-1}{2} \sum_{j=1}^{M} \log |\widehat{\mathbf{C}}_{jj}|.$$

We have explicitly written the parameters that can be chosen to further optimize the lower bound (30), namely the support inputs $\{\mathcal{X}_m^k\}$, and the hyper-parameters $\boldsymbol{\theta}$ which are composed of $\{\boldsymbol{\theta}_k\}$ and $\{\tilde{\boldsymbol{\theta}}\}$ in $K_k$ and $\widetilde{K}$ respectively.

By calculating derivatives of (30) we can optimize the lower bound using a gradient based method. It is easy to see that the complexity for calculating the derivative of

the second and third terms of (30) is $\mathcal{O}(N)$. Thus, the key computational issue of deriving a gradient descent algorithm involves computing the derivative of $\log \Pr(\breve{\boldsymbol{y}}|k)$. We first show how to calculate the inverse of the $N \times N$ matrix $\boldsymbol{\Upsilon} = \boldsymbol{\Lambda}_k \mathbf{C}_{kk}^{-1} \boldsymbol{\Lambda}_k^T + \widehat{\mathbf{C}}^k$. Using the matrix inversion lemma (the Woodbury identity), we have

$$(\boldsymbol{\Lambda}_k \mathbf{C}_{kk}^{-1} \boldsymbol{\Lambda}_k^T + \widehat{\mathbf{C}}^k)^{-1} = [\widehat{\mathbf{C}}^k]^{-1} - [\widehat{\mathbf{C}}^k]^{-1} \boldsymbol{\Lambda}_k \left( \mathbf{C}_{kk} + \boldsymbol{\Lambda}_k^T [\widehat{\mathbf{C}}^k]^{-1} \boldsymbol{\Lambda}_k \right)^{-1} \boldsymbol{\Lambda}_k^T [\widehat{\mathbf{C}}^k]^{-1}.$$

Since $\widehat{\mathbf{C}}^k$ is a block-diagonal matrix, its inverse can be calculated in $\sum_j \mathcal{O}(N_j^3)$. Now, $\mathbf{C}_{kk} + \boldsymbol{\Lambda}_k^T [\widehat{\mathbf{C}}^k]^{-1} \boldsymbol{\Lambda}_k$ is an $m_k \times m_k$ matrix where $m_k$ is the number of inducing variables for the $k$-th mean effect. Therefore the computation of (30) can be done in $\mathcal{O}(m_k^3 + \sum_j N_j^3 + N m_k^2)$. Next, consider calculating the derivative of the first term. We have

$$\frac{\partial \Pr(\breve{\boldsymbol{y}}|k)}{\partial \theta_j} = \frac{1}{2} \breve{\boldsymbol{y}}^T \boldsymbol{\Upsilon}^{-1} \frac{\partial \boldsymbol{\Upsilon}}{\partial \theta_j} \boldsymbol{\Upsilon}^{-1} \breve{\boldsymbol{y}} - \frac{1}{2} \mathbf{Tr}(\boldsymbol{\Upsilon}^{-1} \frac{\partial \boldsymbol{\Upsilon}}{\partial \theta_j}),$$

where, by the chain rule, we have

$$\frac{\partial \boldsymbol{\Upsilon}}{\partial \theta_j} = \frac{\partial \boldsymbol{\Lambda}_k}{\partial \theta_j} \mathbf{C}_{kk}^{-1} \boldsymbol{\Lambda}_k^T - \boldsymbol{\Lambda}_k \mathbf{C}_{kk}^{-1} \frac{\partial \mathbf{C}_{kk}}{\partial \theta_j} \mathbf{C}_{kk}^{-1} \boldsymbol{\Lambda}_k^T + \boldsymbol{\Lambda}_k \mathbf{C}_{kk}^{-1} \frac{\partial \boldsymbol{\Lambda}_k^T}{\partial \theta_j} + \frac{\partial \widehat{\mathbf{C}}^k}{\partial \theta_j}.$$

Therefore, pre-calculating $\breve{\boldsymbol{y}}^T \boldsymbol{\Upsilon}^{-1}$ and sequencing the other matrix operations from left to right the gradient calculation for each hyperparameter can be calculated in $\mathcal{O}(N m_k^2)$. In our implementation, we use stochastic coordinate descent, where at each iteration, one coordinate (parameter) is chosen at random and we perform gradient descent on that coordinate.

### 4.3. Evaluating $\log G(\mathbf{Z}, \mathbf{H}, \breve{\boldsymbol{y}})$

In this section, we develop the expression for $\log G(\mathbf{Z}, \mathbf{H}, \breve{\boldsymbol{y}})$.

$$\begin{aligned}
&\log G(\mathbf{Z}, \mathbf{H}, \breve{\boldsymbol{y}}) \\
&= \int \prod_{l=1}^{M} \prod_{p=1}^{K} \left[ \Pr(\tilde{\mathbf{f}}^l | \mathfrak{f}_p, \boldsymbol{y}^l) \right]^{z_{lp}} \prod_{v=1}^{K} \Pr(\mathfrak{f}_v | \boldsymbol{\eta}_v) \times \sum_{j=1}^{M} \sum_{k=1}^{K} z_{jk} \log \left[ \frac{\Pr(\boldsymbol{y}^j | \mathfrak{f}_k, \tilde{\mathbf{f}}^j) \Pr(\tilde{\mathbf{f}}^j)}{\Pr(\tilde{\mathbf{f}}^j | \mathfrak{f}_k, \boldsymbol{y}^j)} \right] d\mathfrak{F} d\widetilde{\mathcal{F}} \\
&= \sum_{j=1}^{M} \sum_{k=1}^{K} z_{jk} \left[ \int \left( \prod_{v=1}^{K} \Pr(\mathfrak{f}_v | \boldsymbol{\eta}_v) \right) \times \Pr(\tilde{\mathbf{f}}^j | \mathfrak{f}_k, \boldsymbol{y}^j) \times \log \left[ \frac{\Pr(\boldsymbol{y}^j | \mathfrak{f}_k, \tilde{\mathbf{f}}^j) \Pr(\tilde{\mathbf{f}}^j)}{\Pr(\tilde{\mathbf{f}}^j | \mathfrak{f}_k, \boldsymbol{y}^j)} \right] d\mathfrak{F} d\tilde{\mathbf{f}}^j \right] \\
&= \sum_{j=1}^{M} \sum_{k=1}^{K} z_{jk} \left[ \int \Pr(\tilde{\mathbf{f}}^j | \mathfrak{f}_k, \boldsymbol{y}^j) \times \left[ \int \Pr(\mathfrak{f}_k | \boldsymbol{\eta}_k) \prod_{v=1, v \neq k}^{K} \Pr(\mathfrak{f}_v | \boldsymbol{\eta}_v) d\mathfrak{F}_{-k} \right] \times \log \left[ \frac{\Pr(\boldsymbol{y}^j | \mathfrak{f}_k, \tilde{\mathbf{f}}^j) \Pr(\tilde{\mathbf{f}}^j)}{\Pr(\tilde{\mathbf{f}}^j | \mathfrak{f}_k, \boldsymbol{y}^j)} \right] d\mathfrak{f}_k \tilde{\mathbf{f}}^j \right] \\
&= \sum_{j=1}^{M} \sum_{k=1}^{K} z_{jk} \left[ \int \Pr(\tilde{\mathbf{f}}^j | \mathfrak{f}_k, \boldsymbol{y}^j) \Pr(\mathfrak{f}_k | \boldsymbol{\eta}_k) \times \log \left[ \frac{\Pr(\boldsymbol{y}^j | \mathfrak{f}_k, \tilde{\mathbf{f}}^j) \Pr(\tilde{\mathbf{f}}^j)}{\Pr(\tilde{\mathbf{f}}^j | \mathfrak{f}_k, \boldsymbol{y}^j)} \right] d\mathfrak{f}_k d\tilde{\mathbf{f}}^j \right] \\
&= \sum_{j=1}^{m} \sum_{k=1}^{K} z_{jk} \log G(\boldsymbol{\eta}_k, \boldsymbol{y}_j)
\end{aligned} \tag{31}$$

where the second line holds because in the sum indexed by $j$ and $k$ all the product measures

$$\prod_{l=1,l\neq j}^{M}\prod_{p=1}^{K}\left[\Pr(\tilde{\mathbf{f}}^l|\mathfrak{f}_p,\boldsymbol{y}^l)\right]^{z_{lp}},$$

are integrated to 1, leaving only the $\Pr(\tilde{\mathbf{f}}^j|\mathfrak{f}_k,\boldsymbol{y}^j)$. Our next step is to evaluate $\log G(\boldsymbol{\eta}_k,\boldsymbol{y}_j)$, we have

$$\begin{aligned}
&\int \Pr(\tilde{\mathbf{f}}^j|\mathfrak{f}_k,\boldsymbol{y}^j)\Pr(\mathfrak{f}_k|\boldsymbol{\eta}_k)\times\log\left[\frac{\Pr(\boldsymbol{y}^j|\mathfrak{f}_k,\tilde{\mathbf{f}}^j)\Pr(\tilde{\mathbf{f}}^j)}{\Pr(\tilde{\mathbf{f}}^j|\mathfrak{f}_k,\boldsymbol{y}^j)}\right]d\mathfrak{f}_k d\tilde{\mathbf{f}}^j\\
&=\int \Pr(\tilde{\mathbf{f}}^j|\mathfrak{f}_k,\boldsymbol{y}^j)\Pr(\mathfrak{f}_k|\boldsymbol{\eta}_k)\times\log\left[\Pr(\boldsymbol{y}^j|\mathfrak{f}_k,\tilde{\mathbf{f}}^j)\Pr(\tilde{\mathbf{f}}^j)\cdot\frac{\Pr(\boldsymbol{y}^j|\mathfrak{f}_k)}{\Pr(\boldsymbol{y}^j|\mathfrak{f}_k,\tilde{\mathbf{f}}^j)\Pr(\tilde{\mathbf{f}}^j|\mathfrak{f}_k)}\right]d\mathfrak{f}_k d\tilde{\mathbf{f}}^j\\
&=\int \Pr(\mathfrak{f}_k|\boldsymbol{\eta}_k)\log\left[\Pr(\boldsymbol{y}^j|\mathfrak{f}_k)\right]d\mathfrak{f}_k=\int \Pr(\mathbf{f}^j|\boldsymbol{\eta}_k)\log\left[\Pr(\boldsymbol{y}^j|\mathbf{f}^j)\right]d\mathbf{f}^j \qquad (32)
\end{aligned}$$

where the last line holds because of the independence between $\tilde{\mathbf{f}}^j$ and $\mathfrak{f}_k$. We next show how this expectation can be evaluated. This is more complex than the single-task case because of the coupling of the fixed-effect and the random effect.

Recall that $\Pr(\mathbf{f}^j|\boldsymbol{\eta}_k)=\mathcal{N}(\mathbf{f}^j|\mathbf{C}_{jk}\mathbf{C}_{kk}^{-1}\boldsymbol{\eta}_k,\mathbf{C}_{jj}^k-\mathbf{C}_{jk}\mathbf{C}_{kk}^{-1}\mathbf{C}_{kj})$. Denote $\widehat{\mathbf{C}}_{jj}^{-1}=\boldsymbol{L}^{\mathrm{T}}\boldsymbol{L}$ where $\boldsymbol{L}$ can be chosen as its Cholesky factor, we have

$$\log\left[\Pr(\boldsymbol{y}^j|\mathbf{f}^j)\right]=-\frac{1}{2}(\boldsymbol{L}\boldsymbol{y}^j-\boldsymbol{L}\mathbf{f}^j)^T(\boldsymbol{L}\boldsymbol{y}^j-\boldsymbol{L}\mathbf{f}^j)+\log\left[(2\pi)^{-\frac{N_j}{2}}\right]+\log\left[|\widehat{\mathbf{C}}_{jj}|^{-\frac{1}{2}}\right].$$

Notice that $\Pr(\boldsymbol{L}\mathbf{f}^j|\boldsymbol{\eta}_k)=\mathcal{N}(\boldsymbol{L}\mathbf{C}_{jk}\mathbf{C}_{kk}^{-1}\boldsymbol{\eta}_k,\boldsymbol{L}(\mathbf{C}_{jj}^k-\boldsymbol{Q}_{jj}^k)\boldsymbol{L}^{\mathrm{T}})$ where $\boldsymbol{Q}_{jj}^k=\mathbf{C}_{jk}\mathbf{C}_{kk}^{-1}\mathbf{C}_{kj}$. Thus,

$$\begin{aligned}
&\log G(\boldsymbol{\eta}_k,\boldsymbol{y}^j)=\mathbb{E}_{[\mathbf{f}^j|\boldsymbol{\eta}_k]}\log\left[\Pr(\boldsymbol{y}^j|\mathbf{f}^j)\right]\\
&=-\frac{1}{2}\left\|\boldsymbol{L}\boldsymbol{y}^j-\boldsymbol{L}\mathbf{C}_{jk}\mathbf{C}_{kk}^{-1}\boldsymbol{\eta}_k\right\|^2-\frac{1}{2}\mathbf{Tr}(\boldsymbol{L}(\mathbf{C}_{jj}^k-\boldsymbol{Q}_{jj}^k)\boldsymbol{L}^T)+\log\left[(2\pi)^{-\frac{N_j}{2}}\right]+\log\left[|\widehat{\mathbf{C}}_{jj}|^{-\frac{1}{2}}\right]\\
&=\log\left[\mathcal{N}(\boldsymbol{y}^j|\boldsymbol{\alpha}_j,\widehat{\mathbf{C}}_{jj})\right]-\frac{1}{2}\mathbf{Tr}\left[(\mathbf{C}_{jj}^k-\boldsymbol{Q}_{jj}^k)\widehat{\mathbf{C}}_{jj}^{-1}\right]
\end{aligned}$$

where $\boldsymbol{\alpha}_j^k=\mathbf{C}_{jk}\mathbf{C}_{kk}^{-1}\boldsymbol{\eta}_k$. Finally, we have

$$\log G(\mathbf{H},\breve{\boldsymbol{y}})=\sum_{j=1}^{m}\sum_{k=1}^{K}z_{jk}\left[\log\left[\mathcal{N}(\boldsymbol{y}^j|\boldsymbol{\alpha}_j^k,\widehat{\mathbf{C}}_{jj})\right]-\frac{1}{2}\mathbf{Tr}\left[(\mathbf{C}_{jj}^k-\boldsymbol{Q}_{jj}^k)[\widehat{\mathbf{C}}_{jj}]^{-1}\right]\right]. \qquad (33)$$

Furthermore, marginalization out $\boldsymbol{\eta}_k$, we have

$$\mathbb{E}_{\phi^*(\boldsymbol{\eta}_k)}\log G(\boldsymbol{\eta}_k,\boldsymbol{y}^j)=\log\left[\mathcal{N}(\boldsymbol{y}^j|\boldsymbol{\mu}_j^k,\widehat{\mathbf{C}}_{jj})\right]-\frac{1}{2}\mathbf{Tr}\left[\mathbf{C}_{jk}\mathbf{C}_{kk}^{-1}(\boldsymbol{\Sigma}_k-\mathbf{C}_{kk})\mathbf{C}_{kk}^{-1}\mathbf{C}_{jk}\widehat{\mathbf{C}}_{jj}^{-1}\right] \qquad (34)$$

### 4.4. Prediction Using the Sparse Model

The proposed sparse model can be used for two types of problems. Prediction for existing tasks and prediction for a newly added task. We start with deriving the predictive distribution for existing tasks. Given any task $j$, our goal is to calculate the predictive distribution $\Pr(f^j(\boldsymbol{x}^*)|\breve{\boldsymbol{y}})$ at new input point $\boldsymbol{x}^*$, which can be written as

$$\sum_{k=1}^{K} \Pr(f^j(\boldsymbol{x}^*)|z_{jk}=1,\breve{\boldsymbol{y}}) \Pr(z_{jk}=1|\breve{\boldsymbol{y}}) = \sum_{k=1}^{K} r_{jk} \Pr(f^j(\boldsymbol{x}^*)|z_{jk}=1,\breve{\boldsymbol{y}}). \tag{35}$$

That is, because $z_{jk}$ form a partition we can focus on calculating $\Pr(f^j(\boldsymbol{x}^*)|\ z_{jk}=1,\breve{\boldsymbol{y}})$ and then combine the results using the partial labels. Calculating (35) is exactly the same as the predictive distribution in the non-grouped case, the derivation in Section 3.2 gives the details. The complexity of these computations is $\mathcal{O}(K(N_j^3+m^3))$ which is a significant improvement over $\mathcal{O}(KN^3)$ where $N=\sum_j N_j$. Instead of calculating the full Bayesian prediction, one can use *Maximum A Posteriori* (MAP) by assigning the $j$-th task to the center $c$ such that $c=\operatorname{argmax}\Pr(z_{jk}=1|\breve{\boldsymbol{y}})$. Preliminary experiments (not shown here) show that the full Bayesian approach gives better performance. Our experiment below is the results of Bayesian prediction.

Our model is also useful for making prediction for newly added tasks. Suppose we are given $\{\boldsymbol{x}^{M+1},\boldsymbol{y}^{M+1}\}$ and we are interested in predicting $f^{M+1}(\boldsymbol{x}^*)$. We use the variational procedure to estimate its partial labels w.r.t. different centers $\Pr(z_{M+1,k}=1|\mathcal{D})$ and then (35) can be applied for making the prediction. In the variational procedure we update the parameters for $Z_{M+1}$ but keep all other parameters fixed. Since each task has small number of samples, we expect this step to be computationally cheap.

## 5. Related Work

Our work is related to (Titsias, 2009) particularly in terms of the form of the variational distribution of the inducing variables. However, our model is much more complex than the basic GP regression model. With the mixture model and an additional random effect per task, we must take into account the coupling of the random effect and group specific fixed-effect functions. The technical difficulty that the coupling introduces is addressed in our paper, yielding a generalization that is consistent with the single-task solution.

The other related thread comes from the area of GP for multi-task learning. Bonilla et al. (2008)proposed a model that learns a shared covariance matrix on features and a covariance matrix for tasks that explicitly models the dependency between tasks. They also presented techniques to speed up the inference by using Nystrom approximation of the kernel matrix and incomplete Cholesky decomposition of the task correlation matrix. Their model, which is known as the linear coregionalization model (LCM) is subsumed by the framework of convolved multiple output Gaussian process (Álvarez and Lawrence, 2011). The work of Álvarez and Lawrence (2011) also derives sparse solutions which are extensions of different single task sparse GP (Snelson, 2006; Quiñonero-Candela and Rasmussen, 2005). Our work differs from the above models in that we allow a random effect for each individual task. As we show in the experimental section, this is important in modeling various applications. If

the random effect is replaced with independent white noise, then our model is similar to LCM. To see this, from (35), we recognize that the posterior GP is a convex combination of $K$ independent GPs (mean effect). However, our model is capable of prediction for newly added tasks while the models in (Bonilla et al., 2008) and (Álvarez and Lawrence, 2011) cannot. Further, the proposed model can naturally handle *heterotopic* inputs, where different tasks do not necessarily share the same inputs. In (Bonilla et al., 2008), each task is required to have same number of samples so that one can use the property of Kronecker product to derive the EM algorithm.

## 6. Experimental Evaluation

Our implementation of the algorithm makes use of the gpml package (Rasmussen and Nickisch, 2010) and extends it to implement the required functions. For performance criteria we use the standardized mean square error (SMSE) and the mean standardized log loss (MSLL) that are defined in (Rasmussen and Williams, 2005). We compare the following methods. The first four methods use the same variational inference as described in Section 4. They differ in the form of the variational lower bound they choose to optimize.

1. **Direct Inference**: use full samples as the support variables and optimize the marginal likelihood. When $K = 1$, the marginal likelihood is described in Section 3 and the predictive distribution is (5).

2. **Variational Sparse GP for MTL (MT-VAR)**: the proposed approach.

3. **MTL Subset of Datapoints (MT-SD)**: a subset $\mathcal{X}_m^k$ of size $m_k$ is chosen uniformly from the input points from all tasks $\breve{\boldsymbol{x}}$ for each center. The hyper-parameters are selected using $\mathcal{X}_m^k$ (the inducing variables are fixed in advance) and their corresponding observations by maximizing the variational lower bound. We call this MT-SD as a multi-task version of SD (see Rasmussen and Williams, 2005, chap. 8.3.2), because in the single center case, this method uses (4) and (5) using the subset $\mathcal{X}_m, \mathcal{Y}_m$ and $\boldsymbol{x}^j, \boldsymbol{y}^j$ as the full sample (thus discarding other samples).

4. **MTL Projected Process Approximation (MT-PP)**: the variational lower bound of MT-PP is given by the first two terms of (30) ignoring the trace term, and therefore the optimization chooses different pseudo inputs and hyper-parameters. We call it MT-PP because in the single center case, it corresponds to a multi-task version of PP (see Rasmussen and Williams, 2005, chap. 8.3.3).

5. **Convolved Multiple Output GP (MGP-FITC, MGP-PITC)**: the approaches proposed in (Álvarez and Lawrence, 2011). For all experiments, we use code from (Álvarez and Lawrence, 2011) with the following setting. The kernel type is set to be `gg`. The hyperparameters, parameters and the position of inducing variables are obtained via optimizing the marginal likelihood using a scaled conjugated gradient algorithm. The support variables are initialized as equally spaced points over the range of the inputs. We set the $R_q = 1$, which means that the latent functions share the same covariance function. Whenever possible, we set $Q$ which, roughly speaking, corresponds to the number of centers in our approach, to agree with the number of

centers. The maximum number of iterations allowed in the optimization procedure is set to be 200. The number of support variables is controlled in the experiments as in our methods.

Three datasets are used to demonstrate the empirical performance of the proposed approach. The first synthetic dataset contains data sampled according to our model. The second dataset is also synthetic but it is generated from differential equations describing glucose concentration in biological experiments, a problem that has been previously used to evaluate multi-task GP (Pillonetto et al., 2010). Finally, we apply the proposed method on a real astrophysics dataset. For all experiments, the kernels for different centers are assumed to be the same. The hyperparameter for the Dirichlet distribution is set to be $\alpha_0 = 1/K$. Unless otherwise specified, the inducing variables are initialized to be equally spaced points over the range of the inputs. To initialize, tasks are randomly assigned into groups. We run the conjugate gradient algorithm (`minimize.m`) on a small subset of tasks (100 tasks each having 5 samples) to get the starting values of hyperparameters of the $\widetilde{\mathcal{K}}$ and $\mathcal{K}$, and then follow with the full optimization as above. Finally, we repeat the entire procedure 5 times and choose the one that achieves best variational lower bound. The maximum number of iterations for the stochastic coordinate descent is set to be 50 and the maximum number of iterations for the variational inference is set to be 30. The entire experiment is repeated 10 times to obtain the average performance and error bars.

### 6.1. Synthetic data

In the first experiment, we demonstrate the performance of our algorithm on a regression task with artificial data. More precisely, we generated 1000 single-center tasks where each $f^j(x) = \bar{f}(x) + \tilde{f}^j(x)$ is generated on the interval $x \in [-10, 10]$. Each task has 5 samples. The fixed-effect function is sampled from a GP with covariance function

$$\mathbf{Cov}[\bar{f}(t_1), \bar{f}(t_2)] = e^{-(t_1-t_2)^2/2}.$$

The individual effect $\tilde{f}^j$ is sampled via a GP with the covariance function

$$\mathbf{Cov}[\tilde{f}^j(t_1), \tilde{f}^j(t_2)] = 0.25e^{-(t_1-t_2)^2/2}.$$

The noise level $\sigma^2$ is set to be 0.1. The sample points $\boldsymbol{x}^j$ for each task are sampled uniformly in the interval $[-10, 10]$ and the 100 test samples are chosen equally spaced in the same interval. The fixed-effect curve is generated by drawing a single realization from the distribution of $\bar{\mathbf{f}}$ while the $\{\mathbf{f}^j\}$ are sampled i.i.d. from their common prior. We set the number of latent functions $Q = 1$ for MGP.

The results are shown in Fig. (2). The top row shows qualitative results for one run using 20 support variables. We restrict the initial support variables to be in $[-7, 7]$ on purpose to show that the proposed method is capable of finding the optimal inducing variables. It is clear that the predictive distribution of the proposed method is much closer to the results of direct inference. The bottom row gives quantitative results for SMSE and MSLL showing the same, as well as showing that with 40 pseudo inputs the proposed method recovers the performance of full inference. The MGP performs poorly on this dataset, indicating that it is not sufficient to capture the random effect. We also see a large computational advantage

over MGP in this experiment. When the number of inducing variables is 20, the training time for FITC (the time for constructing the sparse model plus the time for optimization) is 1515.19 sec. while the proposed approach is about 7 times faster (201.81 sec.)[1].

### 6.2. Simulated Glucose Data

We evaluate our method to reconstruct the glucose profiles in an intravenous glucose tolerance test (IVGTT) (Vicini and Cobelli, 2001; Denti et al., 2010; Pillonetto et al., 2010) where Pillonetto et al. (2010) developed an online multi-task GP solution for the case where sample points are frequently shared among tasks. This provides a more realistic test of our algorithm because data is not generated explicitly by our model. More precisely, we apply the algorithm to reconstruct the glucose profiles in an intravenous glucose tolerance test (IVGTT) where blood samples are taken at irregular intervals of time, following a single intravenous injection of glucose. We generate the data using minimal models of glucose which is commonly used to analyze glucose and insulin IVGTT data (Vicini and Cobelli, 2001), as follows (Denti et al., 2010)

$$\begin{aligned} \dot{G}(t) &= -[S_G + X(t)]G(t) + S_G \cdot G_b + \delta(t) \cdot D/V \\ \dot{X}(t) &= -p_2 \cdot X(t) + p_2 \cdot S_I \cdot [I(t) - I_b] \\ G(0) &= G_b, \quad X(0) = 0 \end{aligned} \tag{36}$$

where $D$ denotes the glucose dose, $G(t)$ is plasma glucose concentration and $I(t)$ is the plasma insulin concentration which is assumed to be known. $G_b$ and $I_b$ are the glucose and insulin base values. $X(t)$ is the insulin action and $\delta(t)$ is the Dirac delta function. $S_G, S_I, p_2, V$ are four parameters of this model.

We generate 1000 synthetic subjects (tasks) following the setup in previous work: 1) the four parameters are sampled from a multivariate Gaussian with the results from the normal group in Table 1. of (Vicini and Cobelli, 2001), i.e.

$$\begin{aligned} \boldsymbol{\mu} &= [2.67, 6.42, 4.82, 1.64] \\ \boldsymbol{\Sigma} &= \mathbf{diag}(1.02, 6.90, 2.34, 0.22); \end{aligned}$$

2) $I(t)$ is obtained via spline interpolation using the real data in (Vicini and Cobelli, 2001); 3) $G_b$ is fixed to be 84 and $D$ is set to be 300; 4) $\delta(t)$ is simulated using a Gaussian profile with its support on the positive axis and the standard deviation ($SD$) randomly drawn from a uniform distribution on the interval $[0, 1]$; 5) Noise is added to the observations with $\sigma^2 = 1$. Each task has 5 measurements chosen uniformly from the interval $[1, 240]$ and an additional 10 measurements are used for testing. Notice that the approach in (Pillonetto et al., 2010) cannot deal the situation efficiently since the inputs do not share samples often.

The experiments were done under both the single center and the multi center setting and the results are shown in Fig. 3. The plots of task distribution on the left suggest that one can get more accurate estimation by using multiple centers. For the multiple center case, the number of centers for the proposed method is arbitrarily set to be 3 ($K = 3$) and the number of latent function of MGP is set to be 2 ($Q = 2$) (We were not able of obtain reasonable

1. The experiment was performed using MATLAB R2012a on an Intel Core Quo 6600 powered Windows 7 PC with 4GB memory.

results using MGP when $Q = 3$). First, we observe that the multi-center version performs better than the single center one, indicating that the group-based generalization of the traditional mixed-effect model is beneficial. Second, we can see that all the methods achieve reasonably good performance, but that the proposed method significantly outperforms the other methods.

### 6.3. Real Astrophysics Data

We evaluate our method using the astronomy dataset of (Wang et al., 2010), where a generative model was developed to capture and classify different types of stars. The dataset, extracted from the OGLEII survey (Soszynski et al., 2003), includes stars of 3 types (RRL, CEPH, EB) which constitute 3 datasets in our context. One example of each class is shown in Fig. 4. These examples are densely sampled but some stars have less samples and we simulate the sparse case by sub-sampling in our experiments. In previous work (Wang et al., 2010), we developed a grouped mixed-effect multi-task model that in addition allowed for phase shift of the light measurements. As shown in (Wang et al., 2010), stars of the same type have a range of different shapes and the group structure is useful in modeling this domain. However, for inference, Wang et al. (2010) used a simple approach clipping sample points to a fine grid of 200 equally spaced points, due to the high dimensionality of the full sample (over 18000 points).

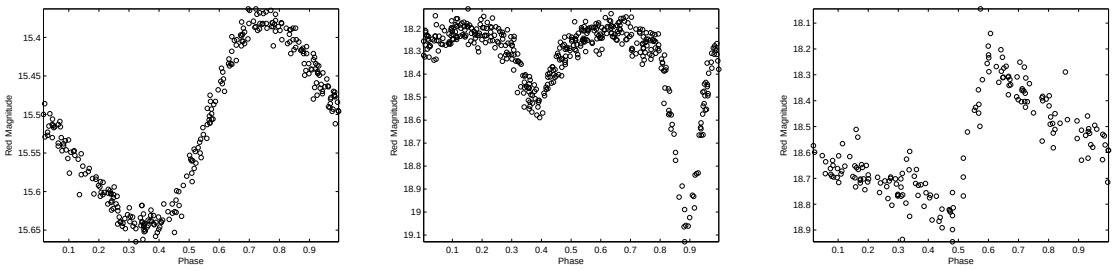


Figure 4: OGLEII: time series for one star (one task in our context) of each type.

Here we use a random subset of 700 stars (tasks) for each type and preprocess the data normalizing each star to have mean 0 and standard deviation 1, and using universal phasing (Protopapas et al., 2006) to phase each time series to align the maximum of a sliding window of 5% of the original points. For each time series, we randomly sample 10 examples for training and 10 examples for testing per evaluation of SMSE and MSLL. The number of centers is set to be 3 for the proposed approach and for MGP we set $Q = 1$ (We were not able to use $Q > 1$). The results are shown in Fig. 5. We can see that the proposed model significantly outperforms all other methods on EB. For Cepheid and RRL whose shape is simpler, we see that the error of the proposed model and MGP are very close and both outperform other methods.

## 7. Conclusion

The paper develops an efficient variational learning algorithm for the grouped mixed-effect GP for multi-task learning, which compresses the information of all tasks into an optimal set of support variables for each mean effect. Experimental evaluation demonstrates the effectiveness of the proposed method. In future, it will be interesting to derive an online sparse learning algorithm for this model. Another important direction is to investigate efficient methods for selection of inducing variables when the input is in high dimensional space. In this case, the clipping method of (Wang et al., 2010) is clearly not feasible, but the variational procedure can provide appropriate guidance.

## Acknowledgement

We would like to thank the authors of Álvarez and Lawrence (2011) who kindly made their code available online. This research was partly supported by NSF grant IIS-0803409. The experiments in this paper were performed on the the Tufts Linux Research Cluster supported by Tufts UIT Research Computing.

## References

M.A. Álvarez and N.D. Lawrence. Computationally efficient convolved multiple output Gaussian processes. *JMLR*, 12:1425–1466, 2011.

M.A. Álvarez, D. Luengo, M.K. Titsias, and N.D. Lawrence. Efficient multioutput Gaussian processes through variational inducing kernels. In *AISTATS*, 2010.

M.A. Álvarez, L. Rosasco, and N.D. Lawrence. Kernels for vector-valued functions: a review. *Arxiv preprint arXiv:1106.6251*, 2011.

J. Bi, T. Xiong, S. Yu, M. Dundar, and R. Rao. An improved multi-task learning approach with applications in medical diagnosis. *ECML*, pages 117–132, 2008.

S. Bickel, J. Bogojeska, T. Lengauer, and T. Scheffer. Multi-task learning for HIV therapy screening. In *ICML*, pages 56–63, 2008.

C.M. Bishop. *Pattern recognition and machine learning*. Springer New York, 2006.

E. Bonilla, K.M. Chai, and C. Williams. Multi-task Gaussian process prediction. *NIPS*, 20: 153–160, 2008.

P. Denti, A. Bertoldo, P. Vicini, and C. Cobelli. Ivgtt glucose minimal model covariate selection by nonlinear mixed-effects approach. *American Journal of Physiology-Endocrinology And Metabolism*, 298(5):E950, 2010.

F. Dinuzzo, G. Pillonetto, and G. De Nicolao. Client-server multi-task learning from distributed datasets. *Arxiv preprint arXiv:0812.4235*, 2008.

A. Gelman. *Bayesian data analysis*. CRC press, 2004.

Michael I. Jordan, Zoubin Ghahramani, Tommi S. Jaakkola, and Lawrence K. Saul. An introduction to variational methods for graphical models. *Machine Learning*, 37, November 1999.

Z. Lu, T.K. Leen, Y. Huang, and D. Erdogmus. A reproducing kernel Hilbert space framework for pairwise time series distances. In *ICML*, pages 624–631, 2008.

G. Pillonetto, G. De Nicolao, M. Chierici, and C. Cobelli. Fast algorithms for nonparametric population modeling of large data sets. *Automatica*, 45(1):173–179, 2009. ISSN 0005-1098.

G. Pillonetto, F. Dinuzzo, and G. De Nicolao. Bayesian Online Multitask Learning of Gaussian Processes. *IEEE T-PAMI*, 32(2):193–205, 2010.

P. Protopapas, J. M. Giammarco, L. Faccioli, M. F. Struble, R. Dave, and C. Alcock. Finding outlier light curves in catalogues of periodic variable stars. *Monthly Notices of the Royal Astronomical Society*, 369:677–696, 2006.

J. Quiñonero-Candela and C.E. Rasmussen. A unifying view of sparse approximate gaussian process regression. *The Journal of Machine Learning Research*, 6:1939–1959, 2005.

C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. The MIT Press, 2005.

C.E. Rasmussen and H. Nickisch. Gaussian Processes for Machine Learning (GPML) Toolbox. *JMLR*, 11:3011–3015, 2010. ISSN 1533-7928.

A. Schwaighofer, V. Tresp, and K. Yu. Learning Gaussian process kernels via hierarchical Bayes. *NIPS*, 17:1209–1216, 2005.

C. Seeger, M. Williams and N. Lawrence. Fast forward selection to speed up sparse gaussian process regression. In *AISTATS 9*. 2003.

Ghahramani Z. Snelson, M. Sparse Gaussian processes using pseudo-inputs. In *NIPS 18*, pages 1257–1264. 2006.

I. Soszynski, A. Udalski, and M. Szymanski. The Optical Gravitational Lensing Experiment. Catalog of RR Lyr Stars in the Large Magellanic Cloud 06. *Acta Astronomica*, 53:93–116, 2003.

M.K. Titsias. Variational learning of inducing variables in sparse gaussian processes. *AISTATS*, 2009.

P. Vicini and C. Cobelli. The iterative two-stage population approach to ivgtt minimal modeling: improved precision with reduced sampling. *American Journal of Physiology-Endocrinology and Metabolism*, 280(1):E179, 2001.

Y. Wang, R. Khardon, and P. Protopapas. Shift-invariant grouped multi-task learning for Gaussian processes. *ECML*, pages 418–434, 2010.

Y. Xue, X. Liao, L. Carin, and B. Krishnapuram. Multi-task learning for classification with Dirichlet process priors. *Journal of Machine Learning Research*, 8:35–63, 2007.

K. Yu, V. Tresp, and A. Schwaighofer. Learning Gaussian processes from multiple tasks. In *ICML*, pages 1012–1019, 2005.

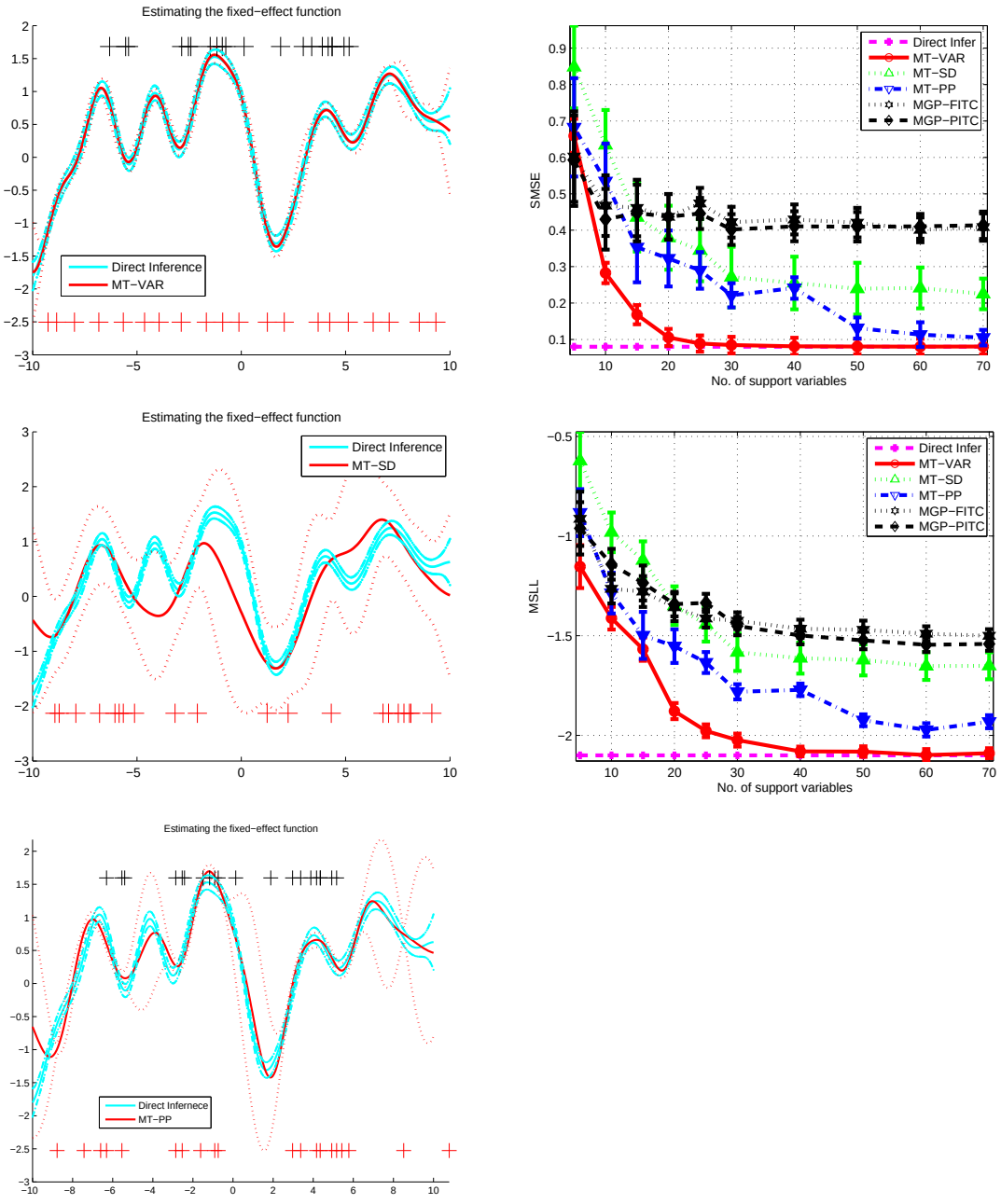


Figure 2: Synthetic Data: Comparison between the proposed method and other approaches. Left Column: Predictive distribution for the fixed-effect. The solid line denotes the predictive mean and the corresponding dotted line is the predictive variance. The black crosses are the initial value of the inducing variables and the red ones are their values after learning process. Right Column: The average SMSE and MSLL for all the tasks.

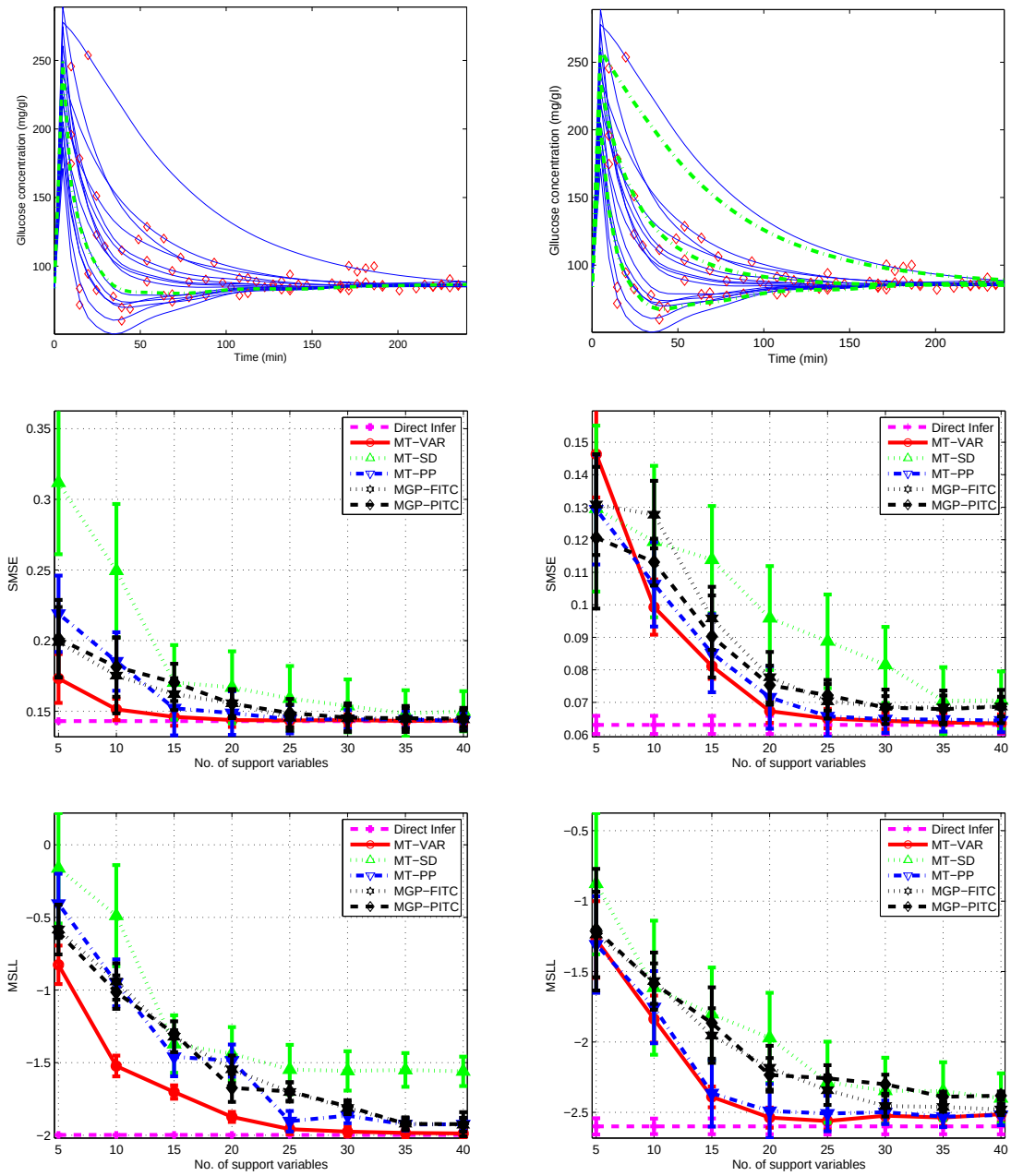


Figure 3: Simulated Glucose Data. Left Column: Single center $K = 1$ results; Right Column: Multiple center $K = 3$ results; Top: 15 tasks (Blue) with observations (Red Diamonds) and estimated fixed-effect curve (Green) obtained from 1000 IVGTT responses. Although the data is not generated by our model, it can be seen that different tasks have a common shape and might be modeled using a fixed effect function plus individual variations. Middle: The average SMSE for all tasks; Bottom: The average MSLL for all tasks.

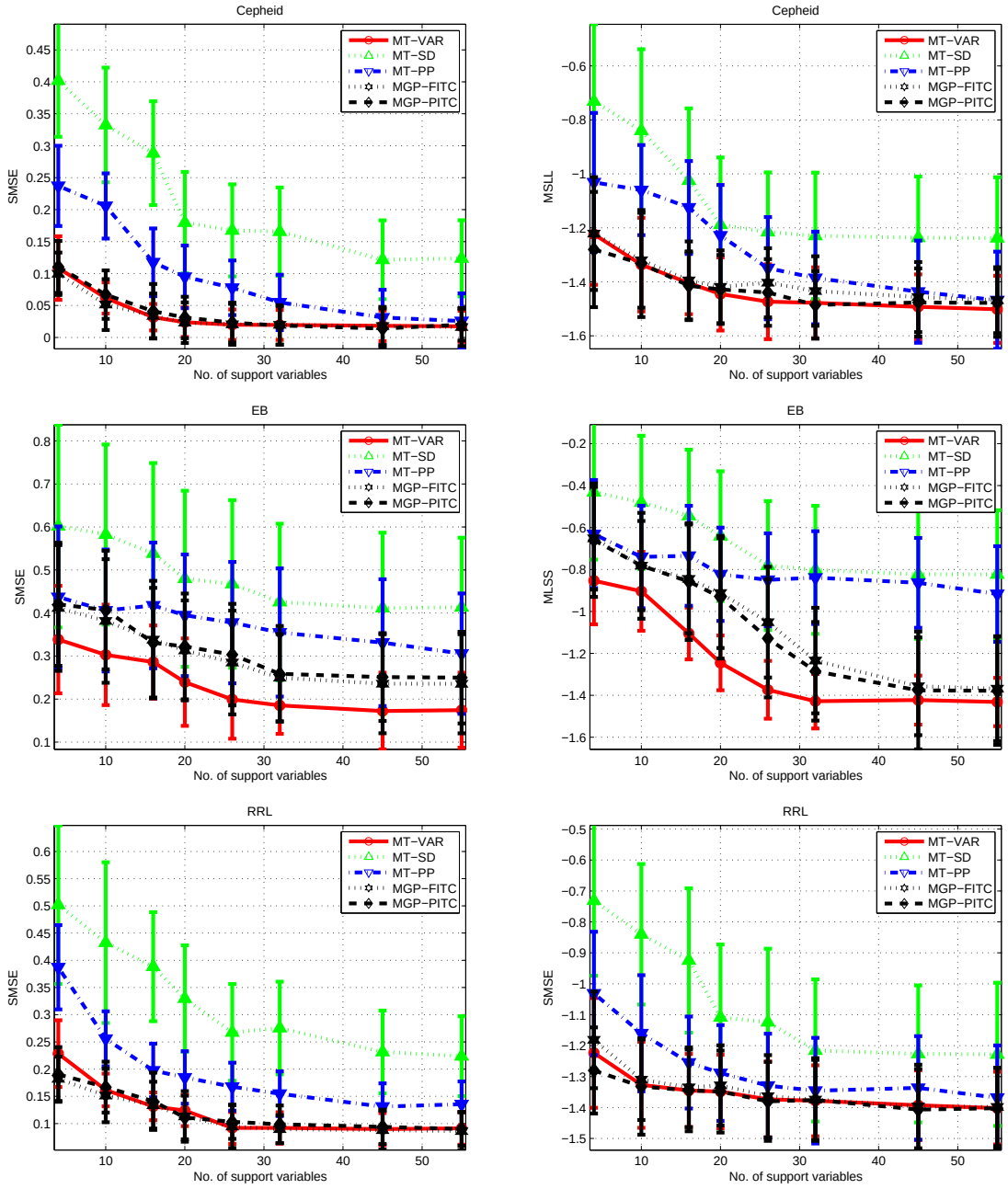


Figure 5: OGLEII: The average SMSE and MSLL for all the tasks are shown in the Left and Right Column. Top: Cepheid; Middle: EB; Bottom: RRL.